



Università di Pisa
Facoltà di scienze matematiche, fisiche e naturali
Corso di laurea specialistica in tecnologie informatiche

**UN SOFTWARE INNOVATIVO DI SUPPORTO
ALL'INSEGNAMENTO DELLA LETTO-SCRITTURA PER
BAMBINI CON DISTURBI DI APPRENDIMENTO**
di Michele Rubino

9 Dicembre 2005

Relatori:
Paolo Mancarella
Maria Rita Laganà
Anna Maria Chilosì

Controrelatore:
Vincenzo Gervasi

Collaboratori esterni: Fondazione IRCCS “Stella Maris”

Anno accademico 2004/2005

*A colei che un giorno mi chiamerà
con il mio vero nome: Amore.*

Ringraziamenti:

Ringrazio la prof.ssa Laganà, che per prima ha sostenuto un'idea difficilissima da far attecchire in un Dipartimento di informatica; il prof. Mancarella per la sua strabiliante disponibilità (senza la quale non sarei andato lontano) e la sua inesauribile voglia di fare, grazie alla quale segue progetti e impegni insostenibili per un informatico qualsiasi; la dott.ssa Chilosì che mi ha insegnato come fare didattica (e credetemi, insegnare a me è più difficile che a un bambino); la dott.ssa Barbara Cerri, che ha mostrato per la mia idea l'entusiasmo che mi ha motivato a continuare e che ha apportato al progetto delle importantissime competenze sulla didattica; Marcello Carrozzino, che non solo mi ha trasformato da un informatico qualunque in un informatico felice di sperimentare e di giocare con suoni, grafica e fantasia (insieme agli altri componenti del laboratorio PERCRO), ma mi ha anche fornito una insostituibile assistenza e collaborazione nel corso di tutto lo sviluppo dell'applicazione di gioco; Alessandra Zoccarì che non solo mi ha introdotto nel mondo della riabilitazione linguistica e mi ha illuminato sulle moderne tecniche didattiche per insegnare lettura e scrittura, ma che è anche una fenomenale ballerina di tango; Carlo Aliprandi e Angelo Paglino, che mi hanno permesso di sperimentare il fascino del riconoscimento vocale per i giochi (anche se non lo immaginano); i miei parenti, che mi hanno sostenuto nel corso di tutta l'università in tutti i modi possibili e desiderabili, soprattutto negli imprevisti; Mimosa Ronca, la mia collega preferita, senza la quale l'università sarebbe stata una perdita di cinque anni della mia vita.

Elenco abbreviazioni e simboli e glossario

3D Studio MAX: uno dei più diffusi programmi per costruire modelli tridimensionali al computer e per animarli.

AAM: formato di file per modelli tridimensionali, utilizzato da ExtremeVR (vedi).

ActiveX: standard definito da Microsoft per la condivisione di informazione fra applicazioni diverse.

Ambiente di sviluppo: applicazione (vedi) che facilita la scrittura di un programma in un dato linguaggio di programmazione, offrendo funzionalità di controllo automatico della sintassi e semplificando le operazioni di compilazione (trasformazione del codice in una forma eseguibile dalla macchina) e di debugging (rilevamento e correzione degli errori).

Applet Java: piccola applicazione (vedi) scritta in linguaggio Java (vedi) che può essere inserita in una pagina HTML (vedi) e può essere scaricata automaticamente e visualizzata tramite un browser (vedi).

AVI: Audio Video Interleave, formato di file standard di Microsoft per i filmati

Applicazione: programma per computer destinato all'uso da parte degli utenti e non alla gestione del computer.

Browser HTML: applicazione (vedi) che permette di visualizzare pagine HTML (vedi), attività principale della navigazione attraverso Internet.

C++: linguaggio di programmazione object-oriented (vedi) fra i più diffusi. La sua caratteristica più rilevante è quella di richiedere al programmatore come deve essere gestita la memoria del computer durante tutto lo svolgimento del programma.

Cella: segmento quadrato del pavimento nel mondo tridimensionale dell'editor (vedi).

Codice (o codice sorgente): testo, scritto in un linguaggio di programmazione, che descrive il comportamento di un programma.

Componente ActiveX: piccola applicazione (vedi) che condivide i dati secondo lo standard ActiveX (vedi) in modo simile ad un'applet (vedi), ma che ha accesso alle funzioni di base di Windows (vedi)

Cube Map: vista dell'ambiente circostante da parte di un osservatore che si trovi al centro di un cubo ideale di lato infinito. Tecnicamente si tratta di un panorama di sei immagini che, disegnate sulle facce interne di un cubo, danno la sensazione a chi si trova all'interno del cubo di essere circondato dal panorama

Deficit fonologico: condizione inferiore rispetto alla norma delle abilità fonologiche (vedi) in un parlante

Disprassia verbale: difficoltà a programmare la sequenza dei suoni nella parola

Disturbi Specifici del Linguaggio: difficoltà nell'uso del linguaggio, pur in assenza di disturbi neurologici, sensoriali o affettivo-relazionali.

DSL: acronimo per Disturbi Specifici del Linguaggio (vedi)

Editor: applicazione (vedi) che consente la modifica di un documento esistente in formato elettronico (in breve, un file) o la creazione di nuovi documenti

ExtremeVR: ambiente di sviluppo prodotto da VRmedia (vedi), particolarmente adatto allo sviluppo di applicazioni (vedi) grafiche e multimediali.

File Zip: file di archivio che contiene uno o più altri file, compressi perché occupino meno spazio

Fonema: in linguistica, “suono”. L'unità minima che si percepisce, si ricorda e si classifica come autonoma.

Fonologia: la scienza dei fonemi e cioè dei suoni, analizzati e classificati nel quadro delle loro rispettive opposizioni e della struttura organica che ne deriva.

Fonologico: relativo alla fonologia, intesa come sinonimo di fonematica e non di fonetica.

Font: insieme di caratteri disegnati tutti con lo stesso stile. Ad esempio, questo glossario è disegnato con un font, il titolo della tesi è disegnato con un altro font.

GLEW:GL Extension Wrangler, libreria per il caricamento delle estensioni di OpenGL (vedi)

Grafema: l'unità grafica minima non suscettibile di divisione di un sistema alfabetico, cioè ciascuna lettera dell'alfabeto

HTML:Hyper Text Markup Language, linguaggio usato per contrassegnare il testo all'interno di una pagina in modo che, visualizzandola con un browser (vedi), abbia un determinato aspetto.

Info-Zip: programma per la compressione di insiemi di file in archivi Zip (vedi) e per la gestione dei file Zip (vedi).

Internet Explorer: uno dei browser (vedi) più diffusi, incluso fra le applicazioni (vedi) di base di Windows (vedi) per visualizzare le pagine HTML (vedi).

Java: linguaggio di programmazione object-oriented (vedi) fra i più diffusi, rinomato per la semplicità di utilizzo e per la gestione automatica della memoria.

Joystick: periferica, solitamente utilizzata nei giochi, costituita da una barra verticale mobile per impostare una direzione (avanti, indietro, a destra, a sinistra) e uno o più pulsanti “di fuoco” il cui uso cambia a seconda del programma utilizzato.

JPG (o JPEG): Joint Photographics Expert Group. Formato di file per le immagini, molto utilizzato grazie alle sue dimensioni contenute

Letto-scrittura: facoltà di utilizzare il linguaggio scritto, leggendolo o producendolo.

Linux: Interfaccia utente grafica basata sul sistema operativo UNIX, che sta prendendo piede soprattutto nelle università e negli uffici come alternativa a Windows (vedi)

Logopedia: disciplina per l’educazione all’uso delle facoltà linguistiche e per il trattamento delle patologie del linguaggio.

Microsoft Visual Studio .Net: Ambiente di sviluppo prodotto da Microsoft per la realizzazione di applicazioni in C++ (vedi), Visual Basic (vedi) e molti altri linguaggi di programmazione.

Mouse: periferica costituita da un corpo centrale mobile, il cui spostamento imposta una direzione, e uno o più pulsanti il cui uso cambia a seconda del programma utilizzato.

MP3: abbreviazione di MPEG (Moving Picture Experts Group) Layer-3, formato di file audio molto diffuso grazie alle sue dimensioni contenute.

MS-DOS: Abbreviazione di Microsoft Disk Operating System, sistema per la gestione del computer in cui l’utente interagisce con il sistema scrivendo una riga di comandi tramite la tastiera (vedi)

Object-oriented: paradigma di programmazione in cui ogni programma è costituito da un insieme di entità astratte chiamate oggetti, ognuno con le sue proprietà. Il funzionamento del programma è determinato dalle definizioni degli oggetti e dalla loro interazione.

OpenGL: libreria grafica open source (vedi) per sviluppare applicazioni (vedi) con grafica tridimensionale.

Open source: filosofia di distribuzione del software in forma di codice sorgente (vedi), così che l’utente possa liberamente copiarlo o modificarlo

Parser: algoritmo che determina e mostra in forma di albero la struttura sintattica di una stringa

Pedagogia: la scienza che studia l’educazione, i suoi metodi e i problemi educativi

Player: applicazione (vedi) di gioco o di visualizzazione di un file.

Plugin: componente software che, una volta inserito in un programma, ne estende le funzionalità.

PNG: Portable Network Graphics. Formato di file per le immagini progettato per l'utilizzo attraverso internet.

Processing fonologico: elaborazione delle informazioni fonologiche, sia durante la lettura che durante la scrittura.

QT: vedi Trolltech QT.

Qualità CD: qualità di un file audio, paragonabile a quella di un compact disc. In termini tecnici, il suono è campionato a 44,100 kHz ed è in formato stereo a 16 bit

Semantica: la scienza dei significati delle parole e delle frasi

Sintassi: lo studio delle funzioni proprie della struttura della frase (potremmo dire, della sua grammatica)

SLI: acronimo di Specific Language Impairment (vedi).

Specific Language Impairment: traduzione in inglese di Disturbi Specifici del Linguaggio (vedi)

Tecnologie Informatiche: corso di Laurea Specialistica proposto dall'Università di Pisa per proseguire gli studi dopo il conseguimento della laurea triennale in Informatica.

Template: modello già pronto; cambiando i parametri del modello si ottengono documenti o dati diversi.

Trolltech QT: Ambiente di sviluppo prodotto da Trolltech, particolarmente adatto allo sviluppo di applicazioni (vedi) a finestre.

Unzip: programma per la decompressione e la gestione degli archivi Zip (vedi)

VBScript: linguaggio per definire degli script (piccoli frammenti di programma) con una sintassi simile a quella di Visual Basic (vedi) che possono essere inclusi in una pagina HTML (vedi).

Visual Basic: linguaggio di programmazione prodotto da Microsoft per lo sviluppo di applicazioni con interfacce grafiche in tempi brevi.

VRmedia: software house che si occupa dello sviluppo di opere multimediali interattive.

WAV: abbreviazione per Wave. Formato di file audio.

Windows: sistema operativo a con interfaccia grafica a finestre sviluppato da Microsoft.

XML: eXtensible Markup Language. Linguaggio usato per contrassegnare testi così da strutturarli in forma di albero sintattico.

XVR: abbreviazione per ExtremeVR (vedi)

Zip: formato dei file Zip (vedi) e abbreviazione per "file Zip"

Abstract

Quella che vi presento di seguito è una Tesi di Laurea Specialistica in Tecnologie Informatiche, in cui, come dice il titolo, si idéa, si implementa e infine si utilizza un software didattico per l'insegnamento dei meccanismi di reduplicazione sillabica sia per il linguaggio orale che per quello scritto.

Si parte, nel primo capitolo, dall'osservazione di un problema che affligge un segmento significativo della popolazione infantile in Italia: i Disturbi Specifici del Linguaggio, o DSL, di cui si dà una presentazione e una suddivisione in categorie. Nel secondo capitolo si effettua una importante riflessione, già fatta in passato da numerosi Autori: i bambini imparano giocando, o almeno imparano di più con uno stile didattico che sia per loro gradevole e stimoli un processo attivo rispetto a uno stile austero e inconsapevole della necessità di stimolare la fantasia e la metacognizione del bambino.

Il terzo capitolo presenta una tecnica di insegnamento della letto-scrittura basata non sui fonemi bensì sulle sillabe come unità fondamentali, essendo queste più "naturali" sia per il sistema cognitivo che per quello articolatorio del bambino, accompagnandolo nello svolgimento di esercizi di ricombinazione sillabica che inducano un modello in cui la compositività delle sillabe genera parole nuove. A questo punto si passa alla costruzione del software di cui si analizza, nel quarto capitolo, la struttura, e nel quinto l'implementazione: il pacchetto è composto da un'applicazione player, sviluppata con ExtremeVR (l'ambiente di sviluppo di VRmedia), con cui il bambino gioca con le sillabe in una serie di schemi, e un'applicazione editor, con cui l'insegnante può personalizzare gli schemi di gioco o crearne di nuovi, inserendo a piacere immagini, suoni, filmati e script. Alcuni di questi schemi sono presentati al capitolo sei, e nel capitolo sette si verifica infine se l'introduzione di un simile strumento informatico può avere successo nel campo del trattamento dei DSL, traendo alcune conclusioni e proposte di lavoro per il futuro.

Parole chiave: Disturbi Specifici del Linguaggio, DSL, logopedia, metodo sillabico, gioco, letto-scrittura, riabilitazione linguistica, ExtremeVR, XVR, C++, QT, Windows

Abstract

What I'm introducing is a graduation thesis in Computer Technology where, quoting from the title, a didactic software to teach syllabic reduplication processes for both spoken and written language is conceived, implemented and, in the end, used.

We start, in the first chapter, from the observation of a problem afflicting a significant share of the children population in Italy: Specific Language Impairments, or SLI, which are given an introduction and a breakdown into subcategories. In the second chapter we make an important observation, formerly made by many Authors: children learn while they play, or at least they learn more by a didactic style that is charming to them and encouraging of active processing, rather than with one that is unsmiling and unconscious of the need of encouraging child's imagination and metacognition.

The third chapter shows an educational approach to reading and writing based not on phonemes but on syllables as foundation units, them being more "natural" for both the child's cognitive system and the articulatory system, bringing him through syllabic combining exercises to induce a model where the composition of syllables generates new words. So we go on to build software, then analyse, in the fourth chapter, the bundled

artifacts and, in the fifth chapter, its implementation: the package is composed by a player application, developed with ExtremeVR (a development environment by VRmedia), where children play with syllables in a sequence of game maps, and an editor application the teacher can use to customise game maps or create new ones, adding images, sounds, movie clips and scripts at will. Some of these maps are shown in chapter six, and in chapter seven we check out if introducing such a computer tool can be successful in the field of SLI treatment, putting forward some conclusions and future work suggestions.

Keywords: *Specific Language Impairment, SLI, logopaedia, logopedics, syllabic method, game, reading/writing, language rehabilitation, ExtremeVR, XVR, C++, QT, Windows*

Sommario

| | Pagina |
|--|---------------|
| Capitolo 1: Osservazione di un problema | 15 |
| Capitolo 2: Idea di partenza | 19 |
| Capitolo 3: Teoria sottostante | 27 |
| Capitolo 4: Com'è strutturato il software | 32 |
| Capitolo 5: Cenni di implementazione | 41 |
| Capitolo 6: Schemi di esempio | 50 |
| Capitolo 7: Risultati e conclusioni | 60 |
| Bibliografia | 61 |

Capitolo 1: Osservazione di un problema

Ciò che fa dell'informatica la disciplina più influente sulla società del XXI secolo è la sua eccezionale versatilità: non più destinate a pochi eletti, le tecnologie informatiche hanno trovato applicazioni in tutti i campi del sapere umano e in numerose problematiche sociali. L'applicazione dell'informatica alla didattica è assai diffusa: i giochi educativi e, per i più grandi, i corsi di lingue straniere con il supporto del computer, sono stati fra i primi software commerciali destinati agli utenti privati¹.

È stato mentre studiavo le potenzialità del connubio fra informatica e didattica e le interfacce uomo-macchina (ovvero mentre studiavo sia per l'esame di Informatica e Didattica sia per l'esame di Costruzione di Interfacce) che sono venuto a conoscenza della situazione dei bambini con disturbi del linguaggio e dell'apprendimento.

I Disturbi Specifici del Linguaggio (o DSL²) sono alcuni dei più diffusi disturbi dello sviluppo, presenti nella popolazione infantile italiana in una percentuale stimata fra il 3% e il 15% [Caselli&Capirci] [Cipriani et al.]. In generale, un DSL si configura come una difficoltà nell'uso del linguaggio, pur in assenza di disturbi neurologici, sensoriali, affettivo-relazionali e deprivazioni socio-ambientali. [Rapin&Allen] hanno suddiviso l'intero spettro dei DSL in tre grandi gruppi:

- Disturbi misti recettivi-espressivi
- Disordini espressivi
- Disordini nei processi linguistici integrativi di alto livello

Diamo ora una breve spiegazione di questi tre gruppi:

I disturbi misti recettivi-espressivi comprendono l'agnosia verbale (ovvero una forma di sordità sufficientemente grave da impedire di sentire le parole) e i deficit fonologico-sintattici. Tra questi, i deficit del sistema fonologico impediscono l'uso corretto dei suoni linguistici e, nel caso del linguaggio scritto, vanno a intaccare quella parte del modello che costruisce la letto-scrittura alfabetica, fase critica dello sviluppo del linguaggio; una delle più note patologie correlate ai deficit a carico del processing

¹ Ovvero utenti di quello che in inglese è chiamato *home computing*, cioè l'uso del computer che si fa a casa propria e non in ufficio. Giochi didattici per imparare la matematica, la musica, la geografia, la storia e (per i più grandi) la lingua inglese erano in dotazione con parecchi personal computer IBM, Commodore, Apple ed altri.

² In inglese l'acronimo corrispondente è SLI, ovvero Specific Language Impairment

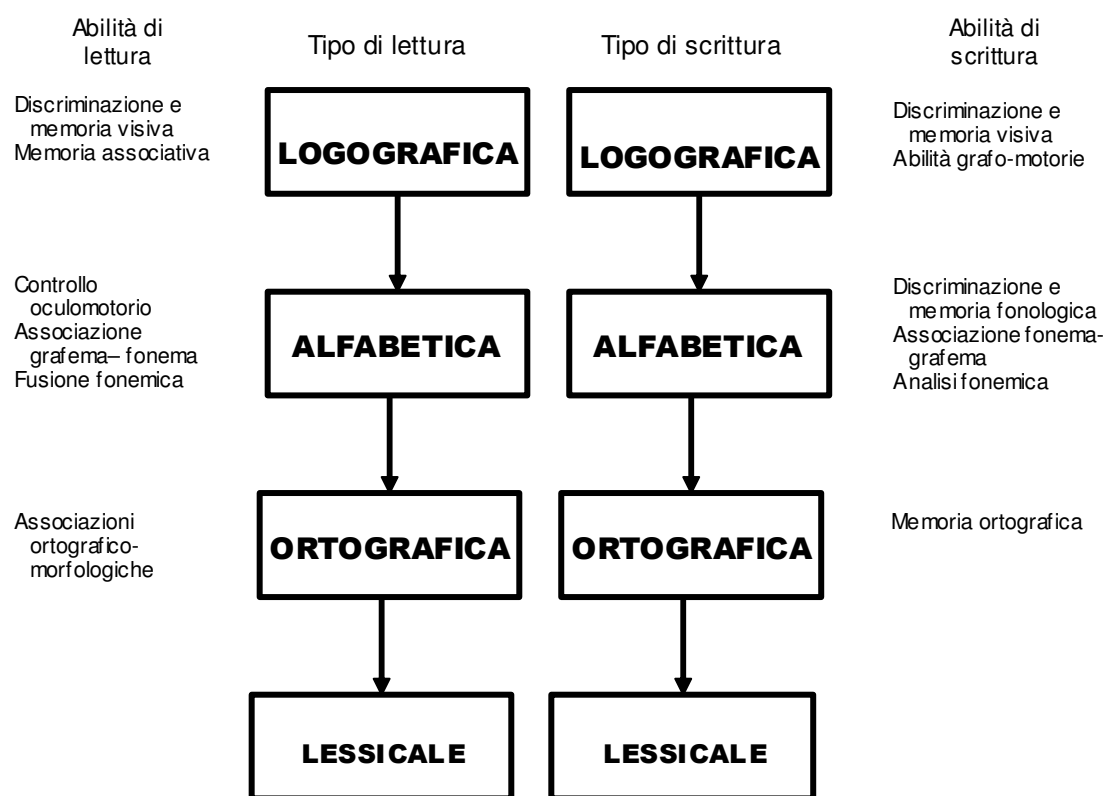
fonologico è la dislessia evolutiva, in molte delle sue manifestazioni³, ma i deficit fonologico-sintattici possono generare una grande varietà di problemi sia in fase di ricezione ed interpretazione del discorso che in fase di produzione del parlato (e dello scritto). I disordini espressivi possono essere caratterizzati da deficit di programmazione fonologica o da una vera e propria disprassia verbale. Entrambe queste patologie sono descritte come difficoltà a programmare la sequenza dei suoni nella parola, con conseguenti errori fonemici, ma mentre nella disprassia questi errori sono sempre diversi, nei deficit di programmazione fonologica compare una certa regolarità negli errori fonemici e la fluenza rimane buona, con una prognosi benigna.

I disordini nei processi linguistici integrativi di alto livello comprendono deficit sintattico-lessicali e deficit semantico-pragmatici. La sindrome da deficit lessicale-sintattico può compromettere la fluenza e il ritmo espressivo, rendendo meno intelligibile e più difficile da produrre il discorso, anche se il sistema fonologico è normale; La sindrome da deficit semantico-pragmatico invece disturba il modo di usare il linguaggio e i contenuti veicolati

Ortogonalmente a questi tre gruppi vale la pena di fare una considerazione sull'età: in [Caselli&Capirci] si osserva che mentre a due anni di età i DSL sono presenti nel 15% dei bambini, oltre i cinque anni di età la percentuale scende al 3%. Spesso perciò non ci si trova di fronte a un DSL persistente, bensì a un "parlatore tardivo"⁴, che recupera spontaneamente il suo ritardo. Come si vede in [Brizzolara et al.], tra il 40 e il 60% dei bambini con DSL in età prescolare, sviluppano in seguito disturbi di apprendimento, soprattutto riguardanti la lingua scritta; tuttavia, ci sono casi in cui le difficoltà nell'apprendimento si presentano anche in soggetti che hanno recuperato il DSL [Bishop&Clarkson]. Per comprendere meglio che cosa siano i DSL e quali disturbi di apprendimento del linguaggio scritto possano verificarsi è opportuno vedere innanzitutto qual è il modello di sviluppo naturale del linguaggio scritto di [Tressoldi] (vedi sotto). Il modello prevede quattro fasi progressive di sviluppo del linguaggio scritto, evidenziate al centro; a lato di ogni fase sono indicate le abilità che permettono di acquisirla:

³ Esistono anche casi di dislessia in cui il deficit è a livello lessicale, e pertanto interferisce con l'automatizzazione della lettura, rallentandola e rendendo difficoltosa la comprensione del testo

⁴ dall'inglese *late talker*



Durante la fase logografica si stabiliscono le associazioni tra una determinata raffigurazione di una intera parola e il suo significato; non c'è generalizzazione sui font (la stessa parola scritta due volte con font diversi viene percepita allo stesso modo di due parole diverse scritte con lo stesso font) né la possibilità di scomporre la parola nei suoi componenti o combinare due parole per formarne una nuova.

Il passaggio alla fase alfabetica è il momento critico dell'acquisizione del linguaggio scritto: il parlante capisce che ci sono delle corrispondenze tra grafemi e fonemi e capisce quali sono queste corrispondenze, ovvero costruisce un alfabeto grafico e un alfabeto fonetico in cui ogni suono è collegato ad un grafema e viceversa, generalizzando anche rispetto a font diversi.

Successivamente il linguaggio arriva a una maturazione con l'instaurarsi di automatismi: è la fase ortografica, in cui alle parole più comuni viene associata una sequenza motoria che permette di pronunciarle o scriverle "in un sol colpo" e un algoritmo di decodifica che legge la parola nella sua interezza e non lettera per lettera; inoltre, nella fase ortografica, le eccezioni del linguaggio (irregolarità dei verbi, irregolarità dei sostantivi nei cambi di genere o numero) vengono sistematizzate e memorizzate. Infine la fase lessicale porta un ulteriore perfezionamento dell'automatizzazione del linguaggio e del sistema delle eccezioni (che, una volta interiorizzate, possono essere previste anche in parole mai sentite prima).

Quando qualcosa interferisce con questo processo di sviluppo nascono i disturbi del linguaggio scritto.

Il discorso sull'età fatto in precedenza è molto importante nell'ambito della rieducazione linguistica e della crescita mentale del bambino: infatti il fattore critico di rischio di veder nascere un disturbo persistente di letto-scrittura nei bambini con DSL è la presenza di un disturbo del linguaggio all'ingresso nella scuola che coinvolge sia il lessico che il processing fonologico. Tuttavia, è possibile recuperare o almeno compensare⁵ i DSL con un'opportuna terapia, ed in particolare con uno stile di insegnamento consapevole del disturbo in atto che aiuti il bambino a superare il suo specifico deficit, assistendolo nell'acquisizione della fase linguistica non adeguatamente perfezionata.

⁵ Per “recuperare” si intende “ripristinare le abilità linguistiche del bambino fino ad un livello paragonabile a quello tipico di uno sviluppo linguistico nella norma”; per “compensare” si intende “fornire al bambino le capacità necessarie a comunicare correttamente e ad apprendere, pur non eliminando del tutto il disturbo”.

Capitolo 2: Idea di partenza

Come ho appena detto, un'educazione adeguata è efficace per evitare che i DSL sorti in età prescolare possano tradursi, al momento dell'ingresso nel sistema scolastico, in difficoltà o ritardi nell'apprendimento del linguaggio scritto. Chiaramente l'educazione al linguaggio scritto deve essere personalizzata per ogni studente, tale è la varietà con cui i disturbi del linguaggio si presentano, e deve essere rispettosa della sua età, delle sue capacità e della sua natura di bambino.

Un gran numero di Autori ha proposto interventi rieducativi per varie tipologie di disturbi; un numero ancora maggiore ha proposto modelli didattici, in base ai quali progettare gli interventi rieducativi, per studenti con e senza disturbi e sono ancora più numerosi gli Autori che hanno discusso sulla teoria pedagogica che deve sottostare ai metodi didattici.

Ci sono però alcuni autori che hanno mostrato un'evidenza empirica condivisa da molti a prescindere dalla loro corrente di pensiero (e presentata in maniera particolarmente gustosa in [Celi&Romani]): i bambini imparano anche quando giocano, e veicolare i contenuti didattici attraverso un gioco può essere più stimolante (sia per gli studenti che per gli insegnanti) ed efficace rispetto ad una lezione frontale⁶, e che più il gioco è attraente maggiore è il suo successo. Visitando la fondazione IRCCS Stella Maris, in provincia di Pisa, ho potuto assistere a qualche sessione di educazione linguistica e ho visto come grazie ad un gioco basato su figure e sillabe più o meno onomatopoeiche ad esse associate il bambino scopriva le proprietà compositive delle sillabe: quando il bambino riusciva ad unire due sillabe, formando così una nuova parola, l'operatore scopriva una nuova figura che corrispondeva alla parola appena generata (e il bambino è contentissimo di veder comparire la nuova figura, segno che il gioco funziona alla grande); gli strumenti utilizzati sono semplicissimi: le figure e le sillabe sono tracciate a penna su un foglio di carta.

Ben sapendo che i bambini già prima di imparare a leggere provano un certo gusto nel maneggiare gli oggetti tecnologici, non era difficile immaginare che i giochi con le sillabe sarebbero stati ancora più attraenti se trasferiti sul computer (per quanto funzionassero benissimo anche con carta e penna) Il computer infatti non offre soltanto la possibilità di utilizzare immagini più colorate, suoni o filmati, ma anche uno speciale concetto di sé: se quello che è dentro il computer è un mondo di gioco, con in mano un

⁶ Una lezione frontale è quella in cui l'insegnante spiega la sua materia mettendosi di fronte agli studenti

joystick (o la tastiera) il bambino diventa un giocatore e un esploratore del mondo. L'ideale sarebbe quindi costruire un mondo, riempirlo di giochi (e di sfide) e lasciare che sia il giocatore a trovare la sua strada. Tuttavia, emergono subito alcuni problemi di quest'idea:

1. Essendo un informatico, le mie competenze non sono adeguate a produrre degli schemi di gioco appropriati per trattare i disturbi dell'apprendimento
2. Come ho detto sopra, è importantissimo utilizzare tecniche di riabilitazione specifiche a seconda del disturbo che si vuole trattare. Un mondo con tutti i giochi possibili per tutti i deficit possibili è impossibile da realizzare, è meno efficace quando si presenta un caso che richiede un trattamento personalizzato differente da quelli già presenti nel mondo e conterrebbe troppi elementi inutili per la didattica, che deve concentrarsi sullo sviluppo delle facoltà deficitarie del bambino.
3. È importante che il bambino interagisca non solo con il computer ma anche con una presenza umana che funga da supervisore, da guida e anche da compagnia, che abbia in sé il calore e l'elasticità (e, riguardo alle terapie, l'esperienza) che è difficile ottenere da un computer

Per il punto 1 la soluzione più semplice è che sia un terapeuta esperto a progettare gli schemi.

Il punto 2 è incentrato sulla personalizzazione a seconda dei casi. La soluzione migliore è lasciare che sia la stessa persona preposta alla diagnosi del disturbo (e che quindi sa in che modo bisogna proporre gli esercizi perché siano efficaci) a mettere a punto gli schemi di gioco personalizzandoli per ogni singolo giocatore.

Il punto 3 si risolve garantendo che ci sia un trainer umano che gioca con il bambino per il tramite del computer e fornendogli l'interfaccia necessaria ad influenzare il mondo di gioco anche durante la sessione al computer e non solo in fase di progettazione degli schemi.

Le soluzioni ai tre problemi mi inducono a proporre non un esercizio o una serie di esercizi, bensì uno strumento che permetta ad un terapeuta di creare da solo i suoi mondi e riempirli dei giochi che più gli sembrano adatti ai bambini con cui si trova a lavorare. La flessibilità dello strumento ne garantirebbe la longevità (gli schemi di gioco sono sempre nuovi), l'efficacia (si può realizzare uno schema apposta per trattare uno specifico caso) e la portabilità fra teorie: terapeuti con differenti approcci teorici

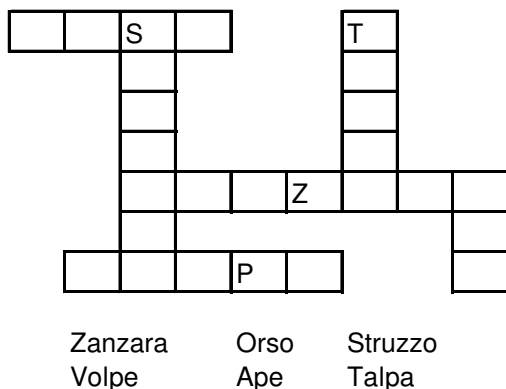
potrebbero comunque utilizzare lo strumento informatico per implementare gli schemi di gioco che si addicano maggiormente al loro modo di fare didattica per renderla più accattivante.

Per rendere più concreto il mio prodotto software ho comunque pensato di sviluppare un insieme di giochi che fungessero da dimostrazione delle capacità dello strumento. L'architettura degli schemi mi è stata fornita dagli operatori del laboratorio di fisiopatologia del linguaggio della Fondazione Stella Maris, che mi hanno anche aiutato a definire le esigenze dei terapeuti; vedremo nel prossimo capitolo qual è l'approccio sottostante gli schemi dimostrativi che ho pensato di accludere al software. Tuttavia, prima di iniziare a costruire schemi di gioco e pezzi di software, ho esaminato il lavoro fatto finora nel campo della rieducazione linguistica e dell'informatica di supporto alla didattica nel trattamento dei disturbi del linguaggio scritto. Molti giochi da tavolo e di enigmistica basati sulle parole sono già utilizzati per attività logopediche:

Memory è un classico gioco di memoria in cui ogni giocatore deve scoprire due carte alla volta fra quelle disposte sul tavolo. Se le due carte sono uguali il giocatore può prenderle, altrimenti le rimette sul tavolo coperte; vince chi ha preso più coppie di carte. Contrassegnando le carte con le parole (anziché, ad esempio, con delle figure) si stimola il giocatore a confrontare la forma grafica delle parole per dire se sono uguali o no e a vederle nella loro interezza anziché scandirle lettera per lettera (esercizio importante per i bambini che hanno difficoltà con la fase ortografica del linguaggio scritto [Tressoldi]). Le regole per formare le coppie possono essere cambiate a seconda dello stimolo che si vuole dare: per esempio, accoppiare le parole di quattro lettere o le parti del corpo contigue come braccio e mano [Judica, Cucciaioni, Verni].

Un gioco molto semplice è quello della lettura di *liste di parole* (da 10 a 20 parole, della stessa categoria, più o meno lunghe) più velocemente possibile.

Più ispirati alle riviste di enigmistica sono gli *incroci di parole* e le *matrici di lettere* [Judica, Cucciaioni, Verni]. Nel primo gioco si mostra al giocatore uno schema di linotipia a chiave:



Il giocatore deve riempire lo schema inserendo in ogni riga (o colonna) una delle parole indicate, aiutandosi nella scelta con la lunghezza delle parole e con alcune lettere facilitatorie già inserite nello schema e rispettando le lettere negli incroci.

Il secondo gioco è complementare al primo, dato che c'è una matrice di lettere nella quale sono contenute alcune parole (scritte orizzontalmente da sinistra a destra, per semplicità), indicate accanto alla matrice.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| A | B | L | F | I | C | H | I |
| T | P | E | S | C | E | H | R |
| P | O | G | I | A | U | S | T |
| V | D | M | A | N | O | E | S |
| M | A | Z | Z | U | R | R | O |
| C | A | L | C | I | O | G | T |
| D | M | R | I | A | N | Q | E |
| U | V | T | R | E | N | O | Z |

| | | |
|-------|-------|---------|
| Treno | Pesce | Calcio |
| Mano | Fichi | Azzurro |

Il giocatore deve individuare le parole nello schema e racchiuderle in un cerchio con la matita.

Le storie figurate e i racconti brevi (da leggere o da produrre) sono uno strumento molto importante per la rieducazione linguistica. Un modo di lavorare con le storie [Judica, Cucciaioni, Verni] è quello di sostituire alcune parole con delle immagini: il bambino denomina le figure man mano che l'insegnante, leggendo la storia, arriva al punto in cui si trova una figura. Successivamente le figure sono sostituite da parole, scritte in rosso, che il bambino deve leggere come succedeva con le figure. Infine il bambino legge tutta la storia.

Vale la pena di notare che questi giochi vanno bene per bambini da 3 a 99 anni (si può quindi giocare con i terapeuti e anche a casa con i genitori, cosa importantissima per tenere alta la motivazione a lavorare con le parole e incoraggiare i rapporti interpersonali) e che alterarne la difficoltà è estremamente semplice: aumentare o ridurre il numero di carte del Memory, cambiare le regole di accoppiamento e usare parole solo della stessa categoria o di più categorie; aumentare o ridurre il numero e la lunghezza delle parole nelle liste di parole; aumentare o ridurre il numero di parole e le dimensioni dello schema nei giochi di incroci di parole e matrici di lettere (e, nel primo dei due, aggiungere o togliere le lettere facilitatorie di partenza); aumentare o ridurre il numero di parole (e figure) nelle storie figurate.

Judica, Cucciaioni, Pollastrini e Verni sono andate ancora oltre, inventando un gioco da tavolo vero e proprio, simile al gioco dell’oca, in cui ogni giocatore, capitando in determinate caselle, deve pescare una carta, che può essere un premio (avanzare di una o due caselle, mangiare una caramella) o un esercizio da superare. Il gioco si chiama «Un gioco di p.a.r.o.l.e» ed è in corso di pubblicazione.

Inoltre, nel campo della riabilitazione linguistica, si fa un uso sempre maggiore di programmi per computer, disponibili in vari tipi a seconda del tipo di lavoro che si vuole svolgere e del rapporto di compresenza fra studenti e insegnanti (didattica per un singolo studente, lavoro di gruppo, esercizi per casa)⁷:

- Alcuni software sono dei semplici strumenti per effettuare test ed esercitare le abilità linguistiche (per esempio *Tachistoscopio*)
- Altri sono “software monotematici”, nel senso che servono ad esercitare una singola abilità (per esempio *Zaffles*, *La battaglia navale* e *Invasori*). Ciò che li distingue da quelli della tipologia precedente è la loro impostazione nettamente più “videogiocosa”, che propone sfide più che esercizi ed è più ricca di grafica, suoni ed interazione, con tanto di ricompensa per i più bravi (monete d’oro, l’affondamento istantaneo di una nave o un posto d’onore nell’agognata classifica dei migliori giocatori).
- Ci sono anche dei programmi con un ampio spettro di tipi di giochi (e di abilità linguistiche da essi esercitate), che funzionano al meglio giocando in gruppo, così da aumentare la motivazione a migliorarsi (soprattutto in ciò in cui si è carenti) per battere i propri amici/compagni di scuola al gioco (un esempio di questi programmi è *Il gioco della papera*).

⁷ Tutti i software nominati come esempio per le tipologie sono descritti più in dettaglio subito dopo.

- Infine esistono dei corsi completi di lettura e scrittura (per esempio *Il giardino della lettura*) progettati per infondere al bambino i rudimenti della dottrina e ad accompagnarlo fino allo sviluppo completo delle sue facoltà.

I giochi descritti sotto sono distribuiti dalla cooperativa [Anastasis](#) e girano sotto MS-DOS o su Windows in modalità MS-DOS.

Il gioco della papera è un gioco dell'oca elettronico, per molti versi simile a «Un gioco di p.a.r.o.l.e», nel senso che, capitando sulle caselle degli esercizi bisogna superarne uno, o rassegnarsi a vedere la propria pedina arretrare di tre caselle. Gli esercizi vertono sul conteggio delle lettere, sul completamento di parola, sui cambi di lettera ed altro ancora.

La battaglia navale segue un principio simile: quando si spara nella casella in cui c'è una nave, perché il colpo vada a segno bisogna risolvere un esercizio; man mano che la nave viene colpita, si scopre che la nave è una parola e che ogni "segmento" di nave è una sillaba! Indovinare la parola che va manifestandosi comporta un vantaggio incredibile, dato che comunicando al gioco la parola scritta sulla nave la si affonda all'istante!

Invasori è un gioco in cui gli invasori spaziali, con i loro dischi volanti, minacciano di sbarcare sulla Terra. Tocca al giocatore respingerli a colpi di contraerea, battendo sulla tastiera la parola scritta sul disco volante per abbatterlo prima che questo riesca ad atterrare (accumulando così punti nella classifica dei migliori punteggi, motivo di orgoglio per videogiocatori di ogni età ed origini). In alternativa, si può giocare con il completamento di parola, battendo la lettera che manca nella parola scritta sul disco volante. La difficoltà può essere modificata a piacere scegliendo la velocità di atterraggio, la lunghezza delle parole, l'intervallo fra la comparsa di due astronavi aliene e il tipo di carattere.

Tachistoscopio è uno strumento molto utilizzato, che permette ad un insegnante di programmare una sequenza di parole, ognuna delle quali viene mostrata solo per qualche istante (si può scegliere un tempo di esposizione di qualche secondo o anche di una frazione di secondo) e poi cancellata dallo schermo, così da spingere l'allievo a velocizzare la lettura e a leggere ortograficamente anziché alfabeticamente.

Le tre applicazioni presentate sotto sono invece distribuite da [Erickson](#) e girano su Windows 95 o superiori.

Start – Avviamento alla lettura è un programma didattico distribuito insieme al libro *Avviamento alla lettura* [Celi, Alberti, Laganà]. Il libro è un testo completo sui sistemi di apprendimento globale e fonetico e il software ne mette in pratica un aspetto: quello dell'associazione immagine-parola con il metodo dell'apprendimento senza errori [Celi&Romani]: inizialmente al bambino vengono mostrate parole associate alle immagini, così da abituarlo alla corrispondenza fra un concetto e la sua forma scritta e, gradualmente, mentre il bambino esplora le “pagine” del programma, ogni volta individuando fra quelle presenti la parola che gli viene chiesta dal computer, le figure si affievoliscono fino a scomparire, lasciando solo la parola a simboleggiare il concetto. Nel caso in cui il bambino cominciasse a mostrare delle difficoltà (sbagliando le risposte alle domande poste dal software), il programma ripristina l'aiuto delle figure, per dargli più tempo di abituarsi alla corrispondenza immagine-parola. Lo stesso tipo di lavoro si può fare anche con le singole lettere, con le immagini che gradualmente cambiano forma fino a diventare lettere stampate (un serpente che man mano perde colori e tratti animaleschi fino a diventare una grande S).

Il giardino della lettura è un corso di lettura⁸ con il metodo dell'apprendimento senza errori (che condivide con *Start* uno degli autori) in cui il bambino può esplorare (cliccando con il mouse) il giardino che costituisce il mondo di gioco. Ad ogni passo del giardino gli viene sottoposto uno stimolo di corrispondenza lettere-parole, reiterato e potenziato dalla voce fuori campo (per esempio “E come Elefante”) o un piccolo esercizio sugli stimoli appena presentati (ad esempio, per l'iniziale di parola, il bambino deve scegliere fra le lettere mostrate dal computer l'iniziale di “Elefante”; nel metodo dell'apprendimento senza errori all'inizio la E è l'unica lettera selezionabile).



Zaffles è un gioco che, come molti altri suoi simili, lavora su un singolo aspetto della letto-scrittura: l'iniziale di parola. Il giocatore deve scoprire la parola che comincia con la lettera misteriosa sottopostagli dal gioco. La lettera ha una forma simile o colori simili alla parola da scoprire (per esempio, una G colorata come il mantello di una giraffa, o una serpeggiante S con tanto di lingua biforcuta e scaglie da rettile) e c'è un editor per disegnare nuove lettere misteriose. Indovinare la parola e scriverla alla tastiera ricompensa l'abile solutore con una borsa di monete d'oro, che possono essere spese acquistando degli “indizi” che lo aiutino ad individuare le parole nascoste.

⁸ Ovvero può essere giocato anche da un bambino che non sia stato ancora esposto al linguaggio scritto.

Il modello di interazione programmato in ognuno di questi software determina anche il modo migliore di utilizzarlo: *Zaffles* ad esempio ha un personaggio guida che spiega al giocatore tutto ciò che gli serve e lo segue nel gioco; l'interazione è molto semplice (basta un click per chiedere un indizio e digitare la parola nascosta per passare allo schema successivo) e dunque è progettato per un singolo giocatore. *La battaglia navale* invece da il suo meglio giocando in due, uno contro l'altro, e allo stesso modo *Il gioco della papera* e *Invasori* sono adatti a gruppi di giocatori in competizione (diretta sul tabellone de *Il gioco della papera*, o indiretta attraverso la classifica dei punteggi di *Invasori*), di cui uno può anche essere l'insegnante (e non è detto che sia il più bravo!); gli strumenti come *Tachistoscopio* prevedono la presenza di un insegnante in carne ed ossa, che prepari in anticipo le sequenze di parole e valuti le prestazioni del bambino. Nei corsi completi la presenza dell'insegnante può essere prevista o meno dal programmatore, ma *Il giardino della lettura* può anche essere esplorato liberamente dal bambino, affidandosi solo alle "voci fuori campo" del programma per orientarsi. È infine giunto il momento di esaminare qual è l'approccio sottostante il mio personale lavoro.

Capitolo 3: Teoria sottostante

Cominciamo facendo una distinzione fra i quattro tipi fondamentali di sistemi di scrittura, ovvero:

1. Il sistema pittografico, in cui ad ogni parola è associato un simbolo, ad esempio  per indicare la parola “alp” (bue)⁹ [Wikipedia].
2. Il sistema logografico (o ideografico), che parte dallo stesso concetto del sistema pittografico ma in cui la stilizzazione è tale che l’ideogramma non ha più alcuna somiglianza con l’immagine a cui è associato. Il carattere kanji (sistema ideografico giapponese) per la parola “drago” (ryuu) è , ben diverso dall’antico pittogramma in cui un rettile (rappresentato dalla forma sulla destra) con quattro zampe e una corona in testa si arrampica su un albero (rappresentato dalla forma sulla sinistra).
3. Il sistema alfabetico, in cui (di norma¹⁰) ogni grafema rappresenta un fonema.
4. Il sistema sillabico, in cui (di norma¹¹) ogni grafema rappresenta una sillaba.

La principale differenza fra i primi due sistemi e i secondi due è che nei sistemi sillabico e alfabetico la parola scritta rappresenta quella orale e non il concetto ad essa sotteso. Un altro dato interessante è la presenza di abilità metalinguistiche come la segmentazione sillabica e la sintesi sillabica in bambini in età prescolare ed in adulti analfabeti; queste abilità sono presenti in varie lingue (tra cui italiano, inglese e giapponese) e sono controllate anche dai cattivi lettori [Emiliani, Furia, Superchi]. La sillaba, d’altra parte, è facilmente percettibile come unità fondamentale nella produzione del parlato, laddove per sviluppare la fase alfabetica del linguaggio è necessario possedere la consapevolezza fonemica [Tressoldi]. Inoltre alcuni studi sugli adulti analfabeti e su adulti letterati cinesi (descritti in [Emiliani, Furia, Superchi]) sembrano dimostrare che la consapevolezza

⁹ Si riesce ad intuire l’evoluzione del pittogramma attraverso vari stadi fino a giungere alla lettera greca α (alpha)

¹⁰ In varie lingue alfabetiche esistono delle ambiguità nella transcodifica fonema-grafema: ad esempio, in italiano, il fonema /k/ può essere scritto con i grafemi C, CH, Q e CQ; viceversa ci sono grafemi come S, che corrisponde ai due fonemi /s/ (/s/ /o/ /l/ /e/) e /z/ (/k/ /a/ /z/ /a/), e addirittura sequenze come GL, che corrisponde al singolo fonema /ʎ/ (come nella parola “moglie”) o alla coppia di fonemi /g/ /l/ (come nella parola “glottide”). Esistono comunque, in lingue come l’italiano e lo spagnolo, regole precise per queste relazioni non-iniettive perciò l’ortografia resta comunque molto regolare, mentre in inglese ci sono molte parole irregolari, in cui le regole di transcodifica non sono rispettate [Emiliani, Furia, Superchi]

¹¹ Ad esempio, esistono delle ambiguità nelle traslitterazioni fra kanji (ideogrammi) e sistemi kana (*hiragana* e *katakana*, sistemi sillabici per la lingua giapponese) [Bilan, Baldwin, Tanaka]

fonemica non è un prerequisito del linguaggio alfabetico, bensì è proprio l'esposizione ad un sistema alfabetico che stimola la nascita nel parlante della consapevolezza fonemica, che una volta arrivata a maturazione gli permette di lavorare sulla fase alfabetica del linguaggio; al contrario, se viene esposto ad un sistema ideografico come il cinese, il parlante non acquisisce la consapevolezza fonemica.

L'ipotesi di lavoro è che, anche se l'italiano è una lingua alfabetica, un metodo di alfabetizzazione basato sulle sillabe potrebbe facilitare alcuni bambini a rischio di disturbi di apprendimento del linguaggio scritto, come quelli in cui, oltre i cinque anni di età, permangono deficit fonologici che impediscono loro di effettuare le segmentazioni e le fusioni fonematiche su cui si basa il metodo fonemico di insegnamento della letto-scrittura. Infatti un metodo sillabico esplicherebbe al bambino «che la parola scritta rappresenta quella orale e che per scrivere occorre operare una qualche segmentazione e conoscere delle corrispondenze fra questa segmentazione e dei segni scritti. [...] È chiaro che, successivamente, si dovrà insistere sulla segmentazione fonemica, ma lo si potrà fare su un sistema già iniziato evitando così la frustrazione del bambino che non riesce ad imparare quello che i compagni stanno facendo» [Emiliani, Furia, Superchi].

[Tressoldi] propone alcune attività di preparazione all'educazione linguistica (per bambini fino alla prima elementare) che stimolano nello studente le abilità metalinguistiche collegate alle sillabe: i primi stimoli sono quello di scomporre le parole proposte dall'insegnante battendo le mani o facendo un salto ad ogni sillaba (segmentazione sillabica) e rispondere ad una domanda presentata in sillabe (fusione sillabica).

Uno dei primi esercizi può essere quello di sintesi sillabica: il bambino deve scegliere fra quattro figure quella che l'insegnante denomina in modo sillabato, alla velocità di circa una sillaba ogni due secondi (da [Emiliani, Furia, Superchi] e [Tressoldi]):



ME_LA

Successivamente si può passare ad un esercizio di individuazione della prima sillaba, che si svolge come quello di sintesi sillabica, ma in cui l'insegnante pronuncia solo la prima sillaba [Emiliani, Furia, Superchi].

Gli esercizi di questi due tipi hanno la loro controparte improntata alla produzione: si presenta al bambino una figura e la parola corrispondente, alla quale però manca la sillaba finale (o iniziale), e gli si chiede di scegliere fra alcune sillabe mostrate in forma di etichette qual è quella che completa la parola. La difficoltà dell'esercizio può essere variata a piacere, riducendo o aumentando il numero di sillabe fra cui scegliere, mettendo nell'insieme di sillabe selezionabili elementi fonologicamente simili o no, fino a richiedere al bambino di costruire l'intera parola (e non soltanto di completarla).



| | |
|-----------|------------|
| MA | ... |
|-----------|------------|

| |
|-----------|
| MO |
|-----------|

| |
|-----------|
| RA |
|-----------|

| |
|-----------|
| NO |
|-----------|

Infine, un esercizio nuovo ideato da Chilosì e Cerri si basa sul principio di ricombinazione sillabica di cui ho parlato nel capitolo 2: si mostra al bambino una figura e una sillaba onomatopeica ad essa collegata, per esempio un treno che fa TU, e gli si chiede di leggerla; per accompagnare il bambino nella soluzione dell'esercizio, inizialmente l'insegnante presenta per tre volte (intervallate da una pausa di un paio di secondi) l'insieme figura-sillaba e pronuncia la sillaba, e solo allora chiede al bambino di provare a leggere la sillaba; Quando il bambino legge la sillaba, si passa ad un'altra figura: un pugno che sbatte sul tavolo, con l'onomatopea TA. Una volta consolidata la corrispondenza tra immagine e sillaba vengono riproposte entrambe le sillabe (con le relative figure) e si chiede al bambino di metterle insieme. Quando il bambino pronuncia TUTA, l'insegnante scopre il disegno della tuta, con la parola TUTA scritta sotto.

L'ipotesi di questo esercizio con supporto iconico e successiva sequenzializzazione dopo memorizzazione e consolidamento della corrispondenza sillaba-parola è quella di sostenere la memoria fonologica di lavoro e il processo di sequenzializzazione che risultano molto spesso carenti nei bambini con DSL [Casalini et al.].



TU



TA



TUTA

L'esercizio però non termina qui: un risultato ancora più importante si ottiene facendo una batteria di fusioni sillabiche come quella esemplificata per la parola "TUTA", in cui ogni sillaba compare in almeno due esercizi. Quando il bambino è in grado di mantenere la giusta corrispondenza e sequenza sillabica questa può essere trasferita anche a parole di lunghezza maggiore (si possono anche usare trisillabi, come MATITA). Vedendo le sillabe che ha già imparato comparire in altre parole e combinarsi in modi diversi facendo scaturire vocaboli sempre nuovi, il bambino scopre la combinatorietà delle sillabe, accedendo alle regole generative per formare nuove parole.

Vedremo nel capitolo 6 come questi esercizi sono stati implementati, ma prima è bene fare conoscenza con l'architettura software che ospita l'esercizio trasformandolo in un gioco da scoprire.

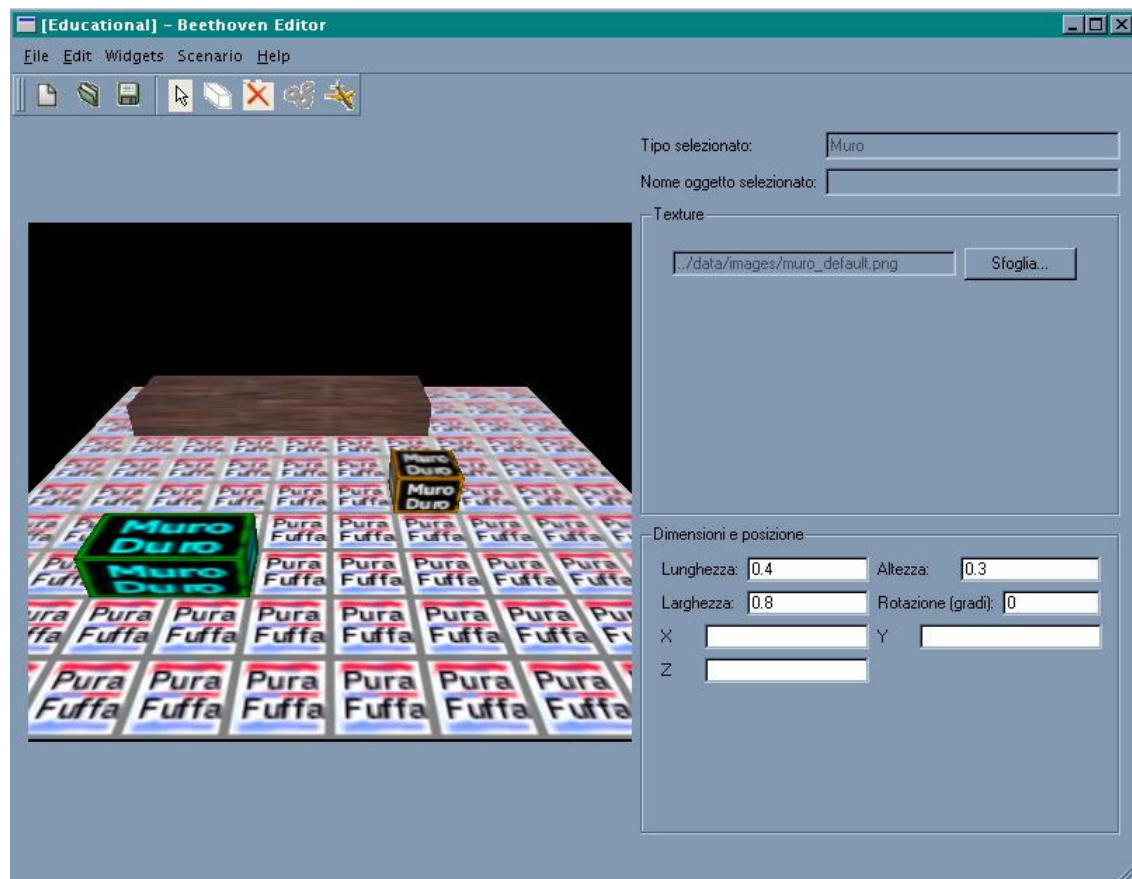
Capitolo 4: Com'è strutturato il software

La combinazione dell'idea di partenza, dell'approccio teorico e della tecnologia informatica si concretizza in un pacchetto software che comprende due applicazioni Windows¹²: un'applicazione **editor** per realizzare gli schemi che possono poi essere giocati con l'applicazione **player**.

L'editor è un'applicazione destinata ai logoterapisti ed agli insegnanti, che ha come scopo principale la realizzazione di schemi di gioco personalizzati per il discente. Dall'osservazione del problema, infatti, risulta una grande varietà di cause della difficoltà nell'apprendimento, e una varietà ancora maggiore di strategie didattiche per trattare queste cause. Come abbiamo visto, attualmente esistono numerose applicazioni didattiche per l'insegnamento di lettura e scrittura, ma queste in gran parte consistono di esercizi già pronti da sottoporre al discente. Può invece capitare che questi esercizi non siano adeguati per il tipo di problema presentato dal discente, o che siano in contrasto con la tecnica d'insegnamento adottata dal docente. Inoltre, una volta completati gli esercizi di una normale applicazione didattica questa esaurisce la sua funzione; occorre perciò proseguire il programma educativo con altri mezzi. La presenza dell'editor, al contrario, assicura al software una grande versatilità e ne aumenta considerevolmente la longevità: dopo aver provato gli esercizi di esempio è possibile definirne altri simili o più difficili, nell'ambito del programma di insegnamento, oppure di realizzare esercizi completamente diversi per trattare un differente tipo di problema di apprendimento. Volendo, è anche possibile andare oltre lo scopo iniziale dell'applicazione e realizzare giochi destinati a studenti più grandi.

La finestra principale dell'editor ha questo aspetto:

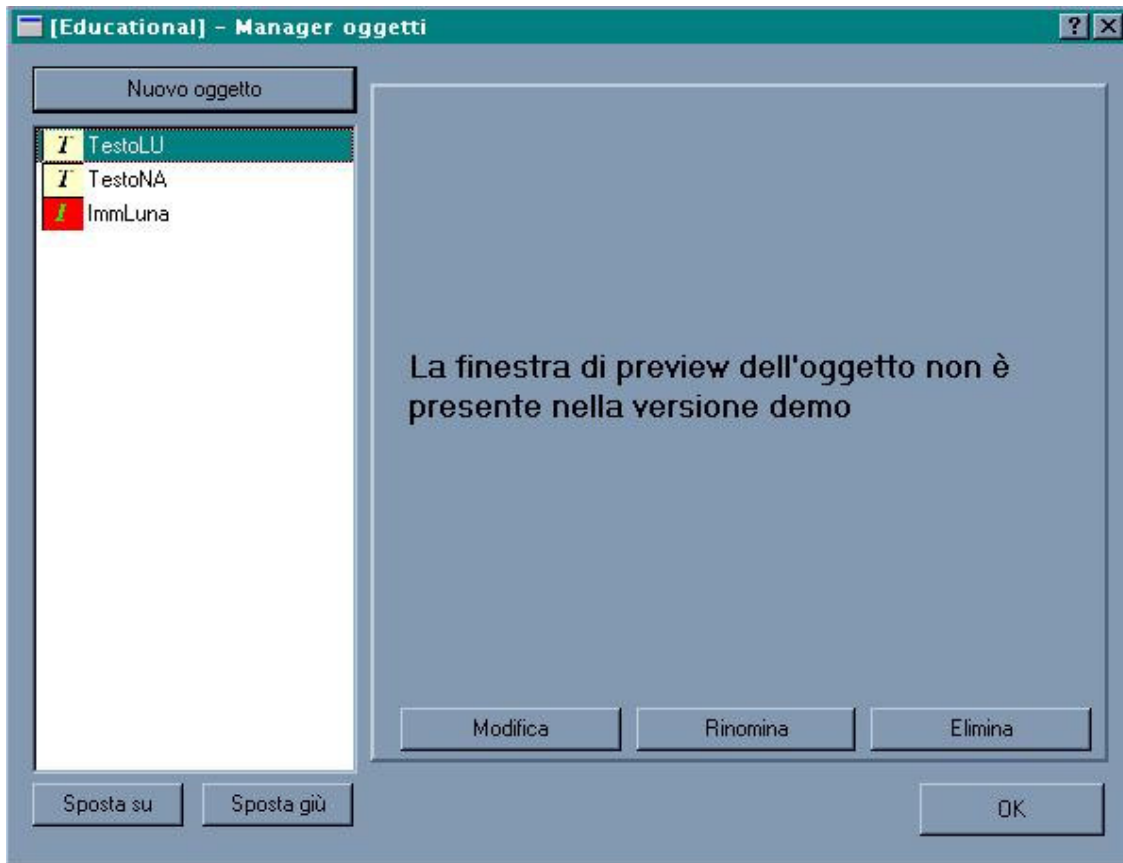
¹² La portabilità su sistemi Linux è fattibile per l'applicazione editor, mentre il player e i componenti ad esso collegati prevedono l'utilizzo di Windows



La parte più importante della finestra è la mappa tridimensionale: il mondo di gioco infatti è in tre dimensioni, ed ha come base un pavimento rettangolare (di dimensioni fissabili dall'utente) suddiviso in celle (sezioni quadrate di pavimento). Il pavimento è la parte di mondo su cui il giocatore può camminare ed esplorare lo schema di gioco; le singole celle (o insiemi di celle) possono essere colorate indipendentemente dalle altre con una texture selezionabile dall'utente, in formato JPG o PNG, ad esempio per riprodurre una spiaggia con il mare o un pavimento di pietra. Sul pavimento è possibile disporre dei muri d'ostacolo per creare un percorso obbligato. I muri possono essere resi alti o spessi a piacere selezionandoli e modificandone le dimensioni attraverso i campi di testo del contenitore *Dimensioni e posizione*; ad esempio un esercizio potrebbe basarsi su una serie di stanze, separate da muri, in ognuna delle quali sia necessario affrontare una prova per passare alla stanza successiva. Anche ai muri, così come alle celle, può essere applicata una texture JPG o PNG.

Uno schema può essere ulteriormente arricchito inserendovi degli oggetti. Gli oggetti possono essere di tre tipi: oggetti testo (visualizzati come scritte sospese a mezz'aria), oggetti modello (modelli 3D in formato AAM) e oggetti immagine/filmato (visualizzati come cartelloni sospesi a mezz'aria e rivolti sempre verso il giocatore su cui è

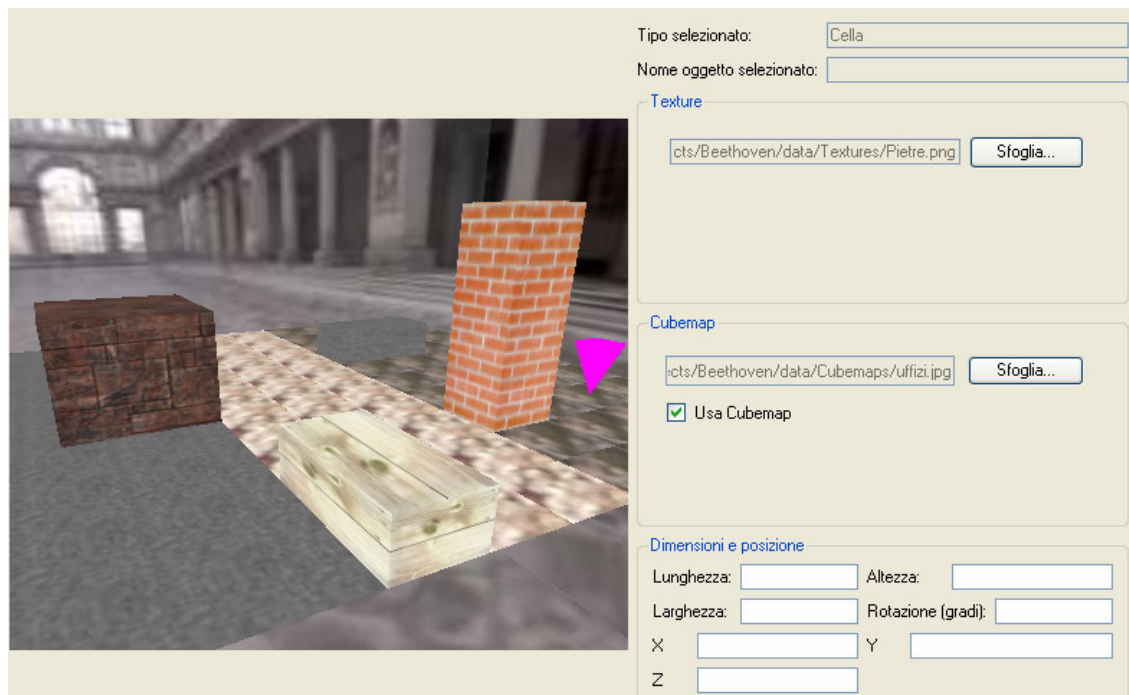
raffigurata un'immagine o proiettato un filmato) e vengono definiti nel Gestore oggetti, la cui finestra viene aperta tramite un comando di menu della finestra principale e ha questo aspetto:



Il gestore degli oggetti, oltre a permettere all'utente di definire nuovi oggetti, funge anche da tavolozza: cliccando su uno degli oggetti nel manager e successivamente su una delle celle nella mappa 3D, un'istanza dell'oggetto comparirà nella mappa. I testi utilizzano un font tipo Century, ad alta visibilità, e possono essere scritti tramite un'apposita finestra di dialogo nel gestore degli oggetti, al momento della creazione; i modelli sono in formato AAM, che è un formato definito da [VRMedia](#), che distribuisce un plugin per 3D Studio MAX per esportare i modelli MAX in formato AAM. Le immagini sono (come sempre) in formato JPG o PNG e i filmati in formato AVI.

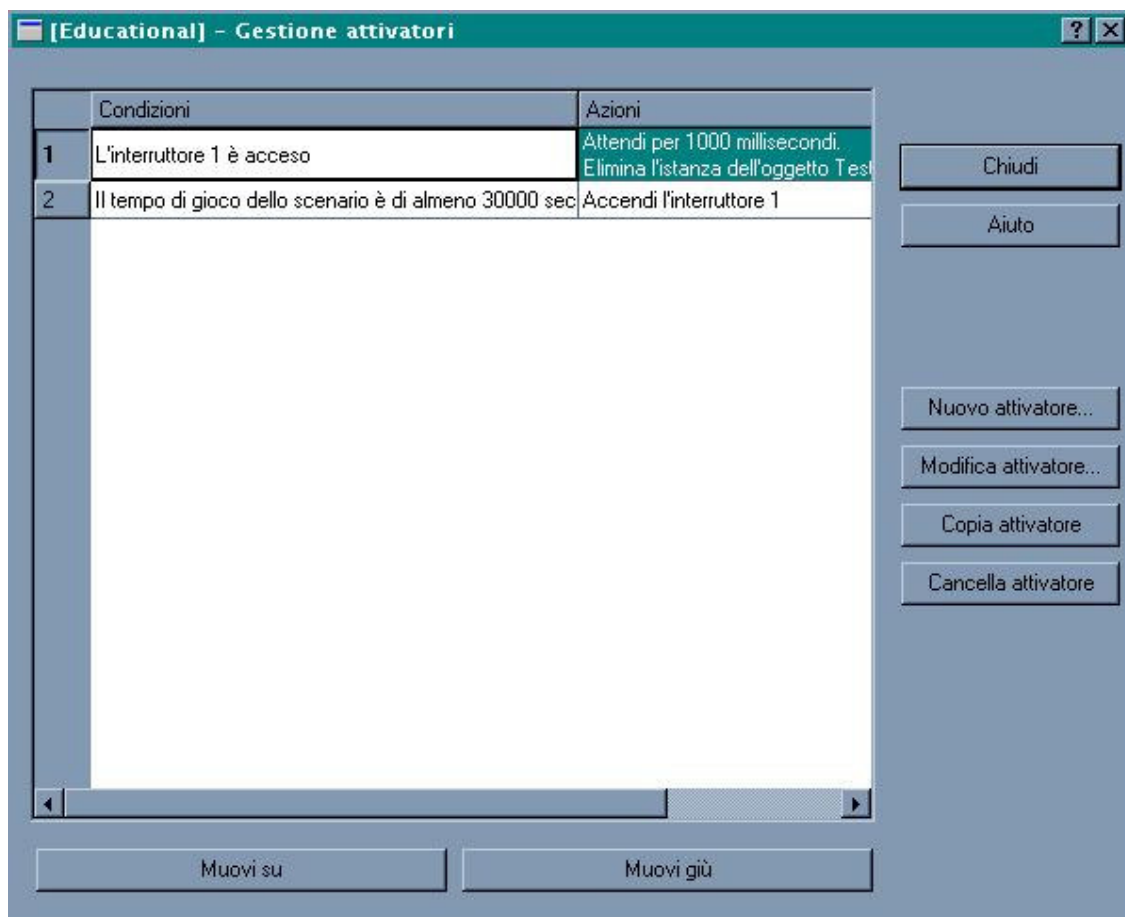
Alla mappa 3D è possibile assegnare anche una Cube Map: questa è una mappa dell'ambiente di sfondo circostante il campo di gioco, ed è composta da sei immagini in formato JPG che idealmente dovrebbero essere collocate sulle sei facce di un cubo al centro del quale si trova il giocatore.

Una volta completato l'allestimento, la mappa assume tutto un altro aspetto:



Per fare uno schema di gioco, però, non è sufficiente una mappa colorata: c'è bisogno che il mondo sia vivo, che chieda l'interazione del giocatore, e reagisca al suo comportamento; per questo esistono gli attivatori. Un attivatore è un piccolo frammento di programma, uno script che, al verificarsi di una o più condizioni definite dall'utente, reagisce eseguendo una o più azioni. Le condizioni possono riguardare la situazione del mondo (tempo di gioco trascorso) o il comportamento del giocatore (posizione del giocatore, clic del mouse, pronuncia di una parola) e le azioni producono un gran numero di cambiamenti nel mondo (creare/spostare/eliminare oggetti, riprodurre suoni sia WAV che MP3, spostare il giocatore).

Gli attivatori sono definibili dall'utente tramite la finestra del Gestore attivatori, anch'essa aperta da un comando di menu:



L'interfaccia per definire e modificare gli attivatori è molto semplice, basata su una serie di finestre di dialogo che via via chiedono all'utente i particolari delle azioni e delle condizioni dell'attivatore. Riprendendo l'esempio delle stanze separate da muri descritto sopra, nel muro potrebbe esserci una porta (un oggetto-modello). Con gli attivatori si può far comparire all'inizio del gioco (quando il tempo di gioco è zero) un oggetto-testo che mostra una parola al centro della mappa e riprodurre un file WAV con una voce fuori campo che dice al giocatore di leggere la parola. Con un altro attivatore, quando il sistema sente la parola pronunciata dal giocatore la porta viene aperta o cancellata, consentendo così il passaggio all'altra stanza. Ed è così che il gioco prende forma...

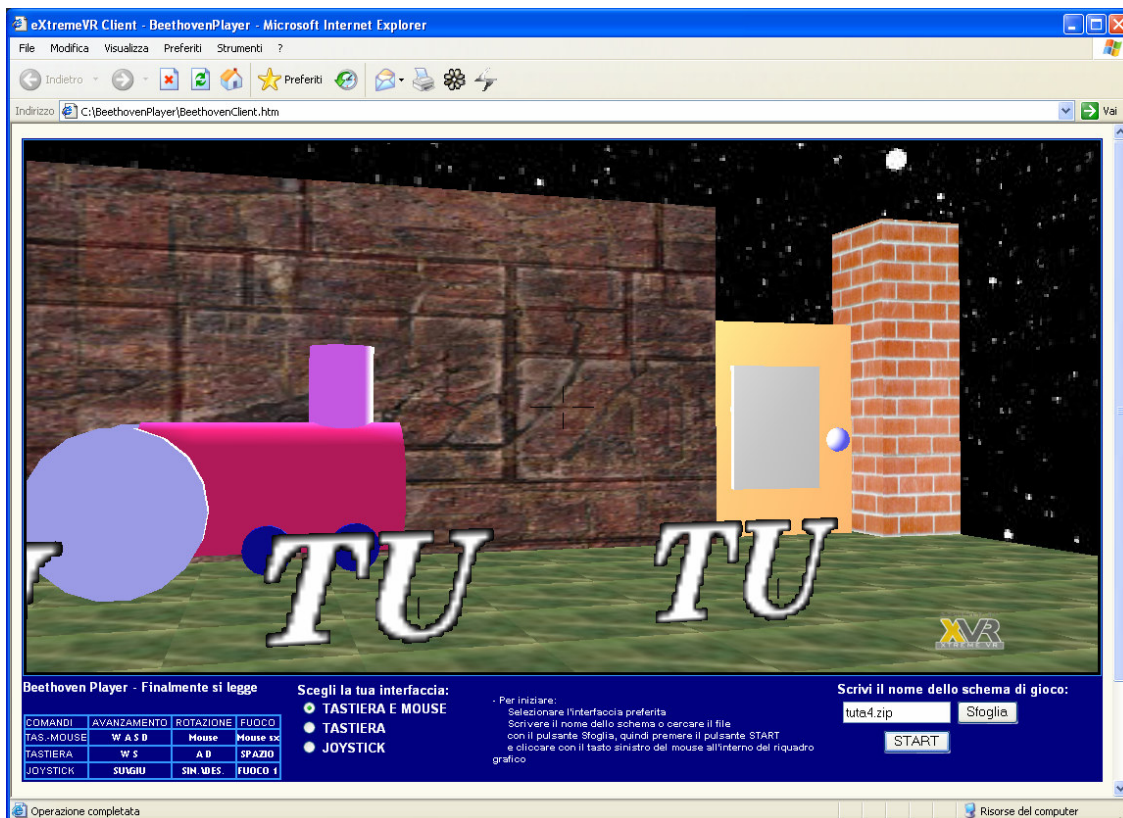
Il codice sorgente dell'editor è in C++, ed è stato sviluppato con [Microsoft Visual Studio .Net](#). L'ambiente a finestre è stato disegnato con [Trolltech QT](#) per Windows. La grafica tridimensionale della mappa 3D sfrutta le librerie OpenGL, con l'aiuto di [GLEW](#) (GL Extension Wrangler). Il file generato dall'editor è un file Zip e può essere aperto con qualsiasi utilità di compressione ed archiviazione; Al suo interno, il file Zip contiene tutti i file grafici per le texture del pavimento, dei muri e della Cube Map, i file sonori, i filmati, i modelli AAM e un file XML in cui c'è la descrizione completa della mappa

tridimensionale (con le associazioni cella-texture, la disposizione dei muri e delle istanze di oggetti), le definizioni degli oggetti e quelle degli attivatori. Il file Zip è generato tramite [Info-Zip](#), un'utilità di compressione open-source, e quando viene aperto nell'editor viene decompresso tramite [Unzip](#) (realizzato dagli stessi autori di Info-Zip).

Il player è l'applicazione destinata agli studenti per giocare gli schemi realizzati dagli insegnanti con l'editor. Il gioco ha una visuale in soggettiva (ovvero il giocatore vede in prima persona, dai suoi "occhi"). Il giocatore parte dal punto di partenza definito nell'editor e cammina sul pavimento dello schema, fra muri e oggetti, usando mouse, joystick o tastiera¹³. Solo giocando uno schema si può avere idea di ciò che si è costruito con l'editor, vedere il testo 3D e le immagini ed osservare il funzionamento degli attivatori (controllati ogni 50 millisecondi).

Il player è realizzato completamente con l'ambiente di sviluppo XVR, prodotto da VRMedia. Il codice sorgente è in un linguaggio di scripting chiamato S3D, che prevede l'emulazione di numerose primitive di OpenGL, consentendo l'utilizzo di uno stile di programmazione simile a quello necessario per utilizzare OpenGL tramite il C++, ma semplificando notevolmente il lavoro grazie a meccanismi come l'inferenza di tipo e a un gran numero di classi e metodi per la gestione di immagini, testo 3D, modelli animati, audio 3D e altro ancora, che sviluppando una normale applicazione C++ richiederebbero di reperire e utilizzare un gran numero di librerie differenti, la cui coesistenza nella stessa applicazione a volte non è semplice da ottenere. XVR genera uno script che, tramite un plugin per Internet Explorer, può essere visualizzato in una pagina HTML:

¹³ Più sotto c'è una descrizione maggiormente accurata delle interfacce di gioco.



In questo esempio, il giocatore si trova su un terreno erboso. C'è un modello di un trenino e degli oggetti testo (le sillabe TU). Un muro di pietra e una colonna di mattoni lo separano dal resto del mondo, ma c'è una porta... Come aprirla? Nel gioco c'è una voce guida a dirlo, ma purtroppo questo è solo uno screen shot¹⁴!

In alto c'è la tipica interfaccia di Internet Explorer, e in basso c'è un'interfaccia blu (realizzata in HTML) che serve a scegliere il file con lo schema e a selezionare l'interfaccia di gioco desiderata fra joystick, tastiera e una combinazione di mouse e tastiera.

Se si utilizza il joystick le direzioni su e giù fanno avanzare o arretrare il giocatore, mentre le direzioni destra e sinistra lo fanno girare verso destra (o sinistra); il movimento con la sola tastiera è simile, solo che “su e giù” sono rimpiazzati dai tasti W ed S, mentre al posto di destra e sinistra ci sono i tasti D e A; la combinazione più interessante di comandi è quella mouse/tastiera: i tasti W e S mantengono la loro funzione, ma A e D invece di far girare il giocatore lo fanno camminare lateralmente a destra e a sinistra, continuando a guardare avanti. Per guardarsi attorno si utilizza il mouse: spostare il puntatore del mouse orienta il giocatore nella stessa direzione, non solo lateralmente ma

¹⁴ Lo stesso screen shot ricomparirà nel capitolo 5, quando si darà una descrizione completa di questo schema di gioco.

anche in alto e in basso, permettendogli di guardare il pavimento o il cielo, secondo uno schema noto a chiunque abbia giocato un moderno shoot'em up in prima persona¹⁵. Le interfacce più utilizzabili sono comunque quelle di joystick e tastiera, dato che tutti gli utenti "tester" hanno mostrato difficoltà nell'uso del mouse per il puntamento. Un altro comando molto importante è il pulsante di fuoco (corrispondente al pulsante 1 del joystick, alla barra spazio o al tasto sinistro del mouse a seconda che l'interfaccia selezionata sia il joystick, la tastiera o la combinazione mouse/tastiera) che simula il clic del mouse su un oggetto per le interfacce alternative definite. Tornando all'interfaccia HTML, il pulsante Sfoglia permette di scorrere facilmente i nomi degli schemi disponibili (ad ogni pressione del pulsante si seleziona il file successivo della lista che il gioco si costruisce automaticamente scorrendo i nomi dei file nella cartella in cui si trova la pagina HTML) e il pulsante START avvia il gioco. Il formato HTML della pagina presenta dei vantaggi non indifferenti:

1. L'applicazione è accessibile dal Web; potrebbe essere installata su un server e gli utenti non dovrebbero fare altro che collegarsi al sito internet che la ospita per giocare.
2. In una pagina HTML si può inserire del codice Visual Basic (VBscript); i tasti Sfoglia e START e l'interazione fra il player e il motore di riconoscimento vocale sono realizzati in questo linguaggio
3. Una pagina HTML può ospitare, oltre al player, componenti ActiveX o applet Java, fungendo da tramite fra questi tipi diversi di software e consentendone l'interazione e l'integrazione.

Il primo punto comporta anche un dettaglio operativo importante: quando il gioco è iniziato, per ricominciare da capo non si riavvia l'applicazione, bensì si ricarica la pagina HTML con il tasto "Aggiorna" o "Reload" (a seconda della lingua del browser).

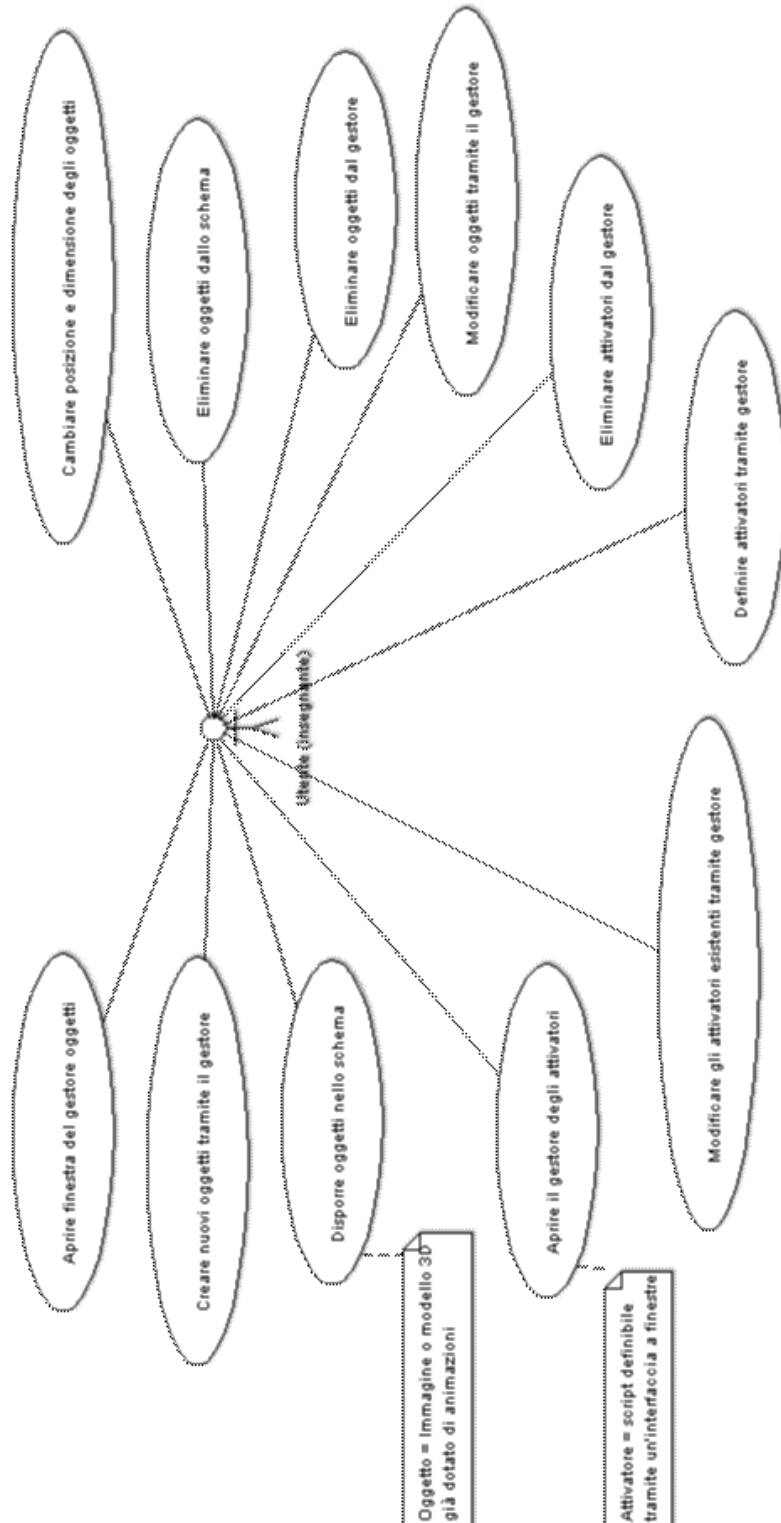
Il secondo e il terzo punto sono stati preziosissimi per l'integrazione nel player del sistema di riconoscimento vocale. Quest'ultimo si concretizza in un componente ActiveX, sviluppato da [SyNTHEMA](#). Il componente si chiama Medical Voice Suite

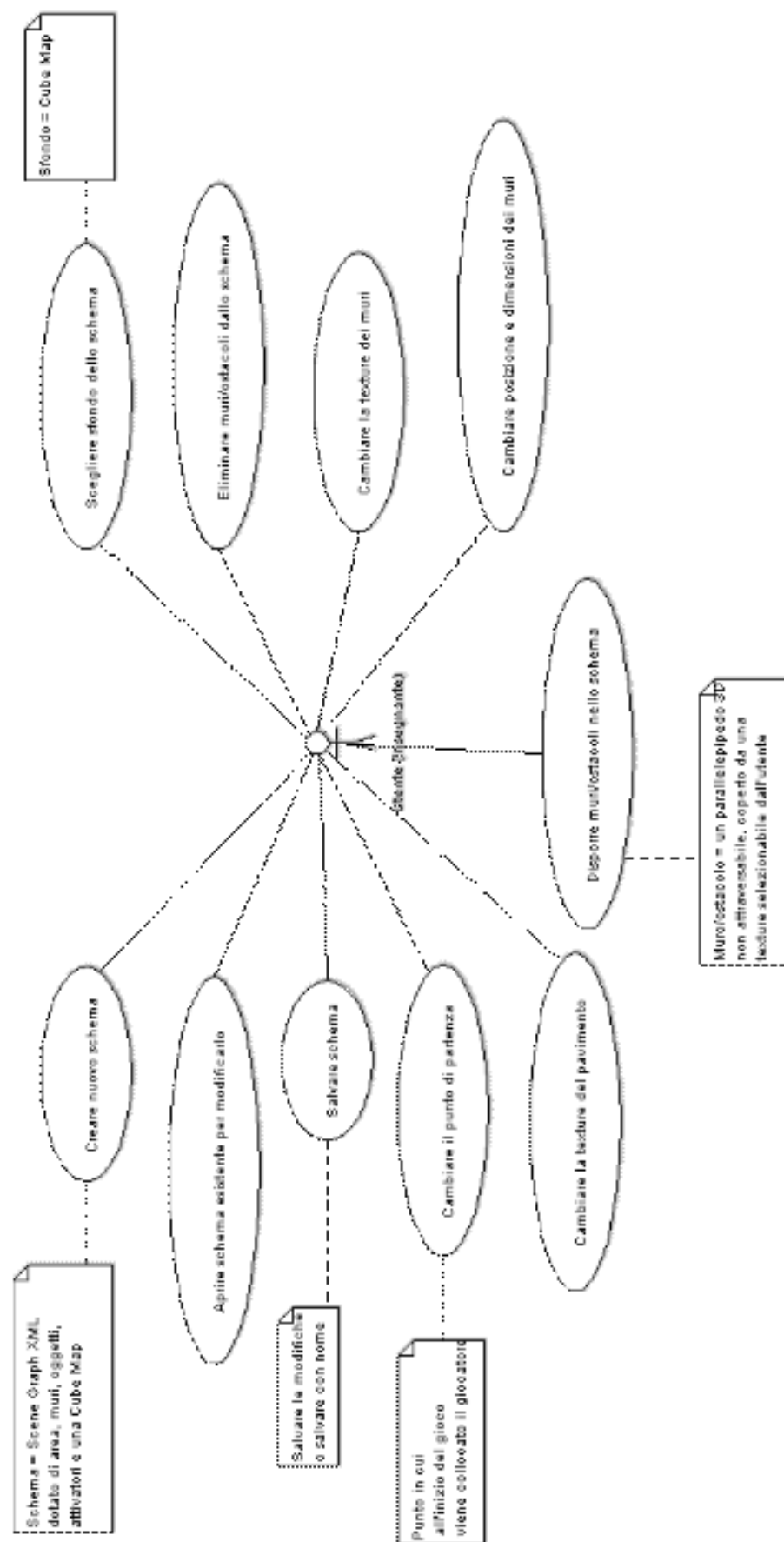
¹⁵ Uno shoot'em up è un gioco in cui il giocatore deve sparare ai propri nemici con armi da fuoco. I nemici sono molto numerosi, ma solitamente dotati di abilità belliche inferiori a quelle del giocatore, e quindi costretti ad affidarsi alla superiorità numerica per prevalere. Riflessi pronti e colpi veloci e precisi sono la chiave per il successo del giocatore. Uno shoot'em up è "in prima persona" quando il giocatore non vede il suo alter-ego elettronico (sia esso un uomo o un mezzo meccanico) dall'esterno, bensì dall'interno, e dunque muove l'alter-ego come se fosse il suo stesso corpo. La combinazione mouse/tastiera è utilizzata in giochi molto noti come Half-Life o in giochi più recenti come Doom 3, Quake 4 o Unreal Tournament 2004

(MVS) e nasce come sistema per la refertazione vocale (ovvero, per far scrivere al computer un referto medico sotto dettatura vocale, senza che il medico debba usare la tastiera) basato su [IBM ViaVoice](#), uno dei sistemi speech-to-text più noti al mondo. Il componente ActiveX è incluso nella pagina HTML e comunica con il player tramite VBscript (il player indica a MVS le parole da riconoscere, MVS invia al player le parole riconosciute). Al momento, tuttavia, il riconoscimento vocale non viene utilizzato, in mancanza di un adeguato addestramento del motore Viavoice perché riconosca le voci dei bambini.

Capitolo 5: Cenni di implementazione

Cominciamo con due diagrammi che mostrano i possibili casi d'uso dell'editor (ci sono due diagrammi per migliorare la leggibilità dei casi d'uso, anche se riguardano entrambi l'editor):





Scendiamo ora più in dettaglio nell'architettura dell'applicazione editor. Per spiegare il modello di interazione fra le classi dell'editor è necessario comprendere la gerarchia degli oggetti in QT. Ogni finestra e ogni elemento grafico e funzionale (liste di elementi, pulsanti, campi di testo, finestre OpenGL) sono detti *widget*. Questi widget formano una struttura gerarchica a seconda di chi li inizializza. Ad esempio, la finestra principale contiene dei pulsanti e dei campi di testo che inizializza all'inizio dell'esecuzione, quindi i pulsanti sono sotto-widget della finestra principale. Anche una finestra come quella del gestore degli attivatori è un sotto-widget della finestra principale dato che viene inizializzata quando l'utente seleziona la voce di menu "Attivatori" della finestra principale.

Un widget può invocare i metodi pubblici (e accedere alle variabili pubbliche) di tutti i suoi sotto-widget, ma non vale il viceversa.

QT offre però anche un altro modo per far interagire i widget: segnali e slot. Ogni widget ha un certo numero di slot, in tutto identici ai metodi normali ma preceduti dal modificatore "slots" nello header (il file .h) della classe. Ad esempio in MainForm.h (header della finestra principale) c'è la dichiarazione

```
public slots:
    virtual void fileNew();
    virtual void fileOpen();
    virtual void fileSave();
    virtual void fileSaveAs();
    virtual void RiceviDataExchange( MessaggioDataExchange * msg );
    virtual void SpedisciDataExchange();
    virtual void SfogliaTexture_clicked();
```

Allo stesso modo si definiscono i segnali, che sono metodi senza corpo. Ad esempio in Scacchiera3DGLWidget.h (header della finestra con la mappa tridimensionale OpenGL) c'è la dichiarazione

```
public:
    //altre intestazioni di metodi

signals:
    void SpedisciDataExchange(MessaggioDataExchange* msg);
```

Segnali e slot sono collegati fra loro con una macro offerta da QT, chiamata Connect. All'inizializzazione della finestra principale (in mainform.ui.h) si connette il segnale della scacchiera3D ad uno slot della finestra principale:

```
connect( scacchiera3DGLWidget2,
        SIGNAL( SpedisciDataExchange( MessaggioDataExchange* ) ),
        this, SLOT( RiceviDataExchange( MessaggioDataExchange * ) ) );
```

Questo significa che, nel momento in cui la scacchiera 3D manda il segnale SpedisciDataExchange, con cui spedisce un messaggio di scambio dati, nella finestra principale si invoca il metodo RiceviDataExchange, passandogli come parametro attuale lo stesso parametro attuale del segnale emesso. Per emettere un segnale, infatti, si utilizza un'altra macro, chiamata emit:

```
MessaggioDataExchange* msg = new MessaggioDataExchange(tipo,
                                                         posizione[X],
                                                         posizione[Y],
                                                         posizione[Z],
                                                         dimensioni[LARGHEZZA],
                                                         dimensioni[LUNGHEZZA],
```

```
dimensioni [ALTEZZA],  
nome);
```

```
emit SpedisciDataExchange(msg);
```

Questo segnale viene mandato dalla scacchiera 3d (in Scacchiera3DGLWidget.cpp) alla finestra principale nel momento in cui l'utente fa clic con il mouse per selezionare un oggetto posizionato nello schema perché la finestra possa mostrare all'utente le dimensioni correnti dell'oggetto, la sua posizione, il nome e il tipo.

Grazie ai segnali ed agli slot, anche un sotto-widget può comunicare con il widget superiore a lui nella gerarchia.

Il prossimo schema è una vista modulare in cui l'editor è decomposto in widget e sotto-widget. C'è da notare che si potrebbe scendere molto più in dettaglio, dato che ogni sotto-widget (per esempio una finestra di dialogo) è a sua volta una combinazione di widget (pulsanti, campi per l'immissione di testo).

Lo schema successivo mostra invece l'interazione fra widget e si vede come i clic del mouse sulla scacchiera 3D (che selezionano o dispongono oggetti, celle e muri) inviano segnali alla finestra principale (MainForm) perché mostri all'utente dimensioni, posizione ed altro, mentre la finestra principale, una volta che i dati sono stati modificati dall'utente (per esempio, cambiando le dimensioni) prepara un messaggio di scambio dati, ma lo passa alla scacchiera 3D semplicemente invocando un metodo.

Allo stesso modo il gestore oggetti, che funge da tavolozza per disporre gli oggetti nella mappa tridimensionale, invia un segnale alla finestra principale quando l'utente ha selezionato un oggetto e la finestra principale invoca un metodo della scacchiera 3D per notificarle di prepararsi a disporre un oggetto.

Una menzione particolare merita l'inizializzazione delle finestre di dialogo che sono sotto-widget di un'altra finestra (il gestore attivatori è sotto-widget della finestra principale, la form nuovo attivatore è sotto-widget del gestore attivatori, la finestra di dialogo nuova condizione è sotto-widget della form nuovo attivatore): subito dopo l'invocazione del costruttore, il widget superiore passa al suo nuovo sotto-widget uno o più puntatori, così da tenere traccia delle modifiche fatte dall'utente anche dopo che il sotto-widget è stato chiuso e deallocato.

I vari widget sono indicati con il nome della loro classe. Ecco una breve legenda delle classi:

MainForm è la finestra principale dell'applicazione.

Scacchiera3DGLWidget è la mappa tridimensionale OpenGL.

DialogChiediSalvataggio chiede all'utente se vuole salvare il file modificato al momento di uscire dal programma o di aprire un altro file o di creare un nuovo file.

QFileDialog apre una finestra Sfoglia per cartelle in puro stile Windows.

FormAttivatori è il gestore degli attivatori, che permette di crearli e modificarli.

FormManagerOggetti è il gestore degli oggetti e permette di crearli e modificarli.

FormNuovoScenario si apre al comando Nuovo file e chiede all'utente le dimensioni della nuova mappa.

DialogNuovoOggetto richiede all'utente nome e valore dell'oggetto appena creato.

DialogTesto serve ad immettere una stringa di testo.

DialogNuovaAzione permette di scegliere tipo e parametri di una nuova azione.

DialogNuovaCondizione permette di scegliere tipo e parametri di una nuova condizione

DialogModificaAzione permette di modificare i parametri di un'azione.

DialogModificaCondizione permette di modificare i parametri di una condizione.

DialogNumDecimale serve ad immettere un numero decimale.

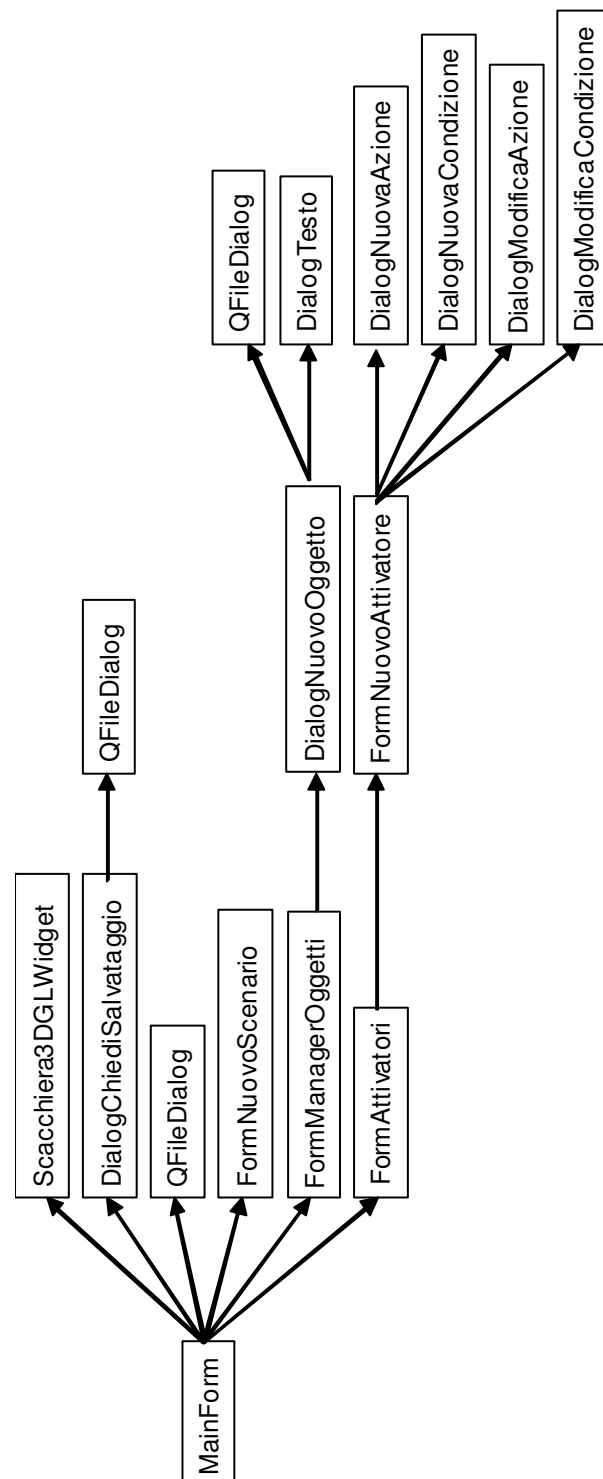
DialogNumIntero serve ad immettere un numero intero.

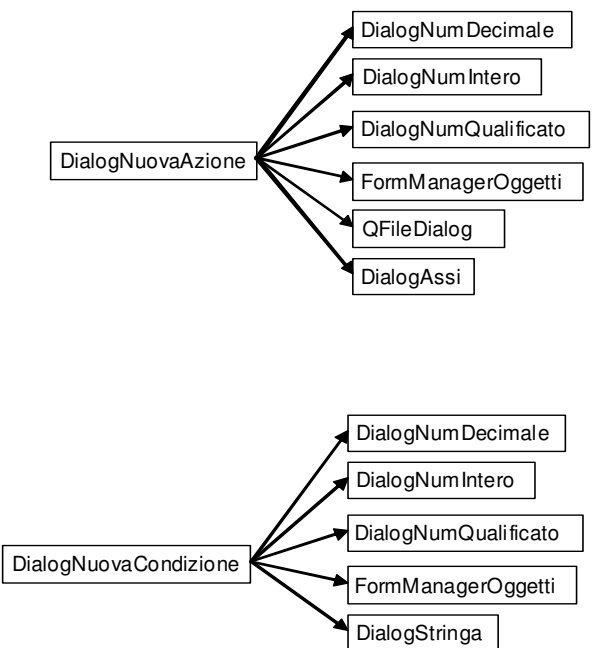
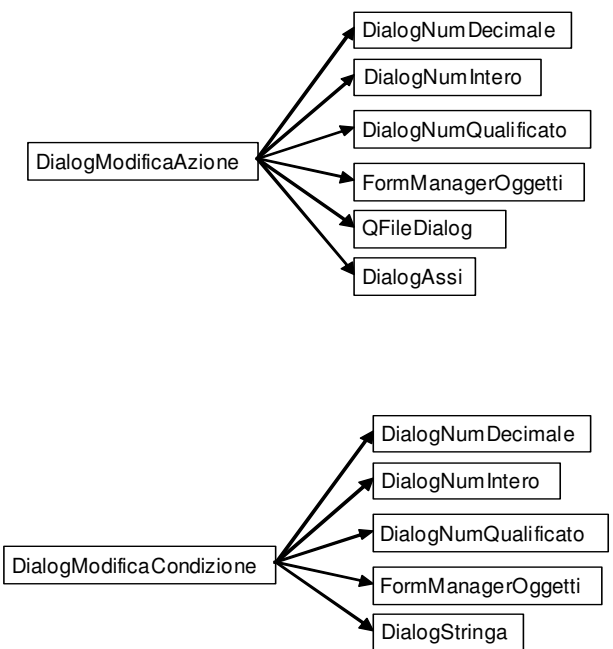
DialogNumQualificato serve ad immettere un numero intero con un qualificatore scelto fra “almeno”, “al massimo” ed “esattamente”.

DialogAssi serve a scegliere un asse cartesiano.

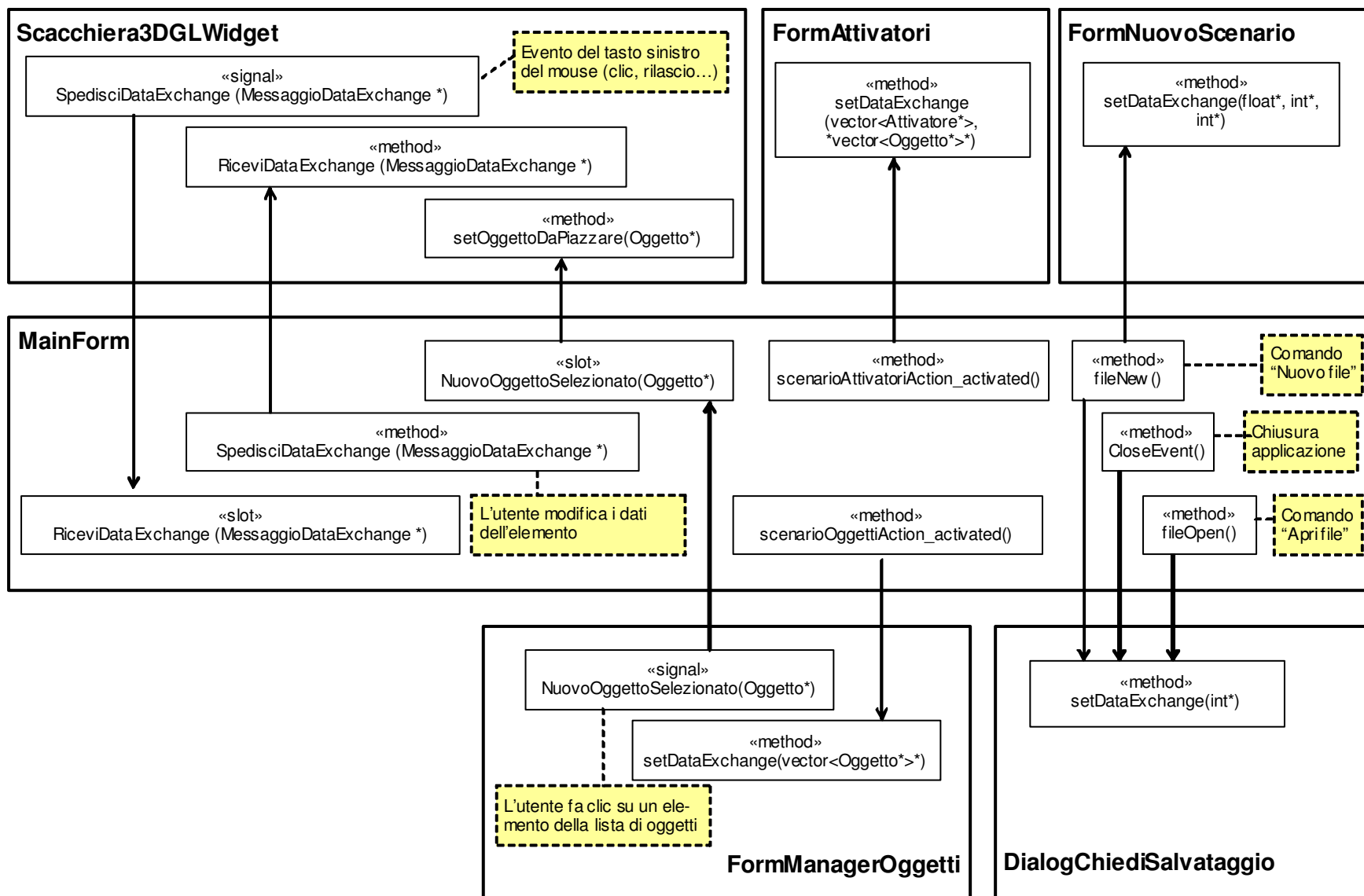
DialogStringa serve ad immettere una breve stringa di testo.

Il diagramma di decomposizione è un grafo diretto aciclico, che ho scomposto in più parti per motivi di leggibilità, in cui la freccia indica la relazione di sotto-widget. Ad esempio, FormNuovoScenario è un sotto-widget di MainForm.



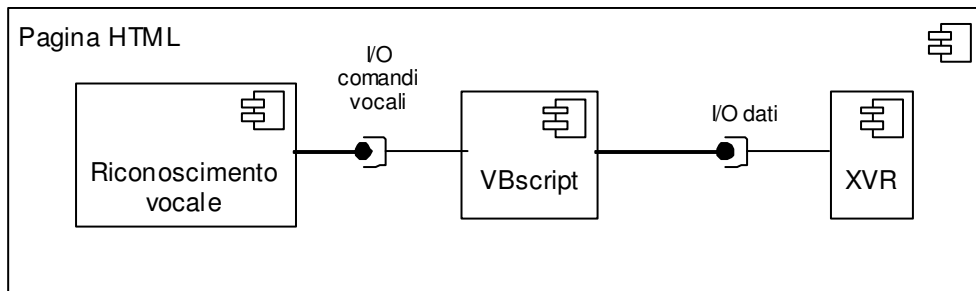


Passiamo alla vista dell'interazione fra i moduli del programma mostrando le connessioni dei metodi, dei segnali e degli slot.

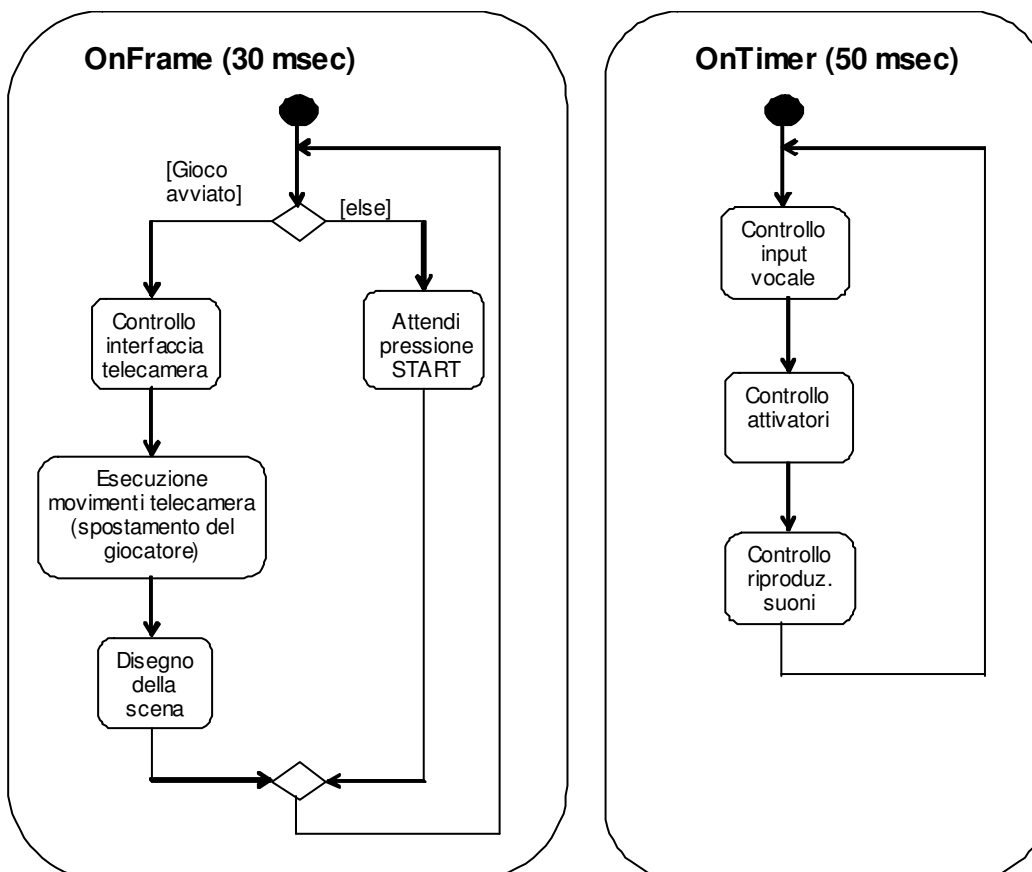


Il player ha una struttura completamente diversa: invece di utilizzare eventi o segnali e slot, ExtremeVR ha una struttura a cicli di controllo per la grafica (ridisegnata 75 volte al secondo), la fisica (controllata 60 volte al secondo), la rete e così via. Io utilizzo solo il ciclo per la grafica (invocazione del metodo OnFrame di ridisegno ogni 30 millisecondi) più uno per controllare gli attivatori (invocazione del metodo OnTimer ogni 50 millisecondi) che funzionano in concorrenza.

Il funzionamento dell'applicazione dipende però dal contesto in cui si trova: è infatti collocata all'interno di una pagina HTML e scambia dati tramite VBscript con il componente di riconoscimento vocale, anch'esso incluso nella pagina HTML:



Vediamo ora i cicli di controllo, a partire dall'avvio (marcato con un pallino nero come stato di ingresso) dell'applicazione:



Il ciclo di disegno controlla sempre se il gioco è stato avviato, ovvero se l'utente ha selezionato un'interfaccia e uno schema ed ha premuto il tasto START. Queste azioni producono il caricamento dello schema, e la OnFrame ha quindi a sua disposizione tutti i dati necessari per disegnare la scena. In più, prima di disegnare, la funzione controlla

l'input da tastiera, mouse e joystick per lo spostamento della telecamera ed effettua lo spostamento desiderato.

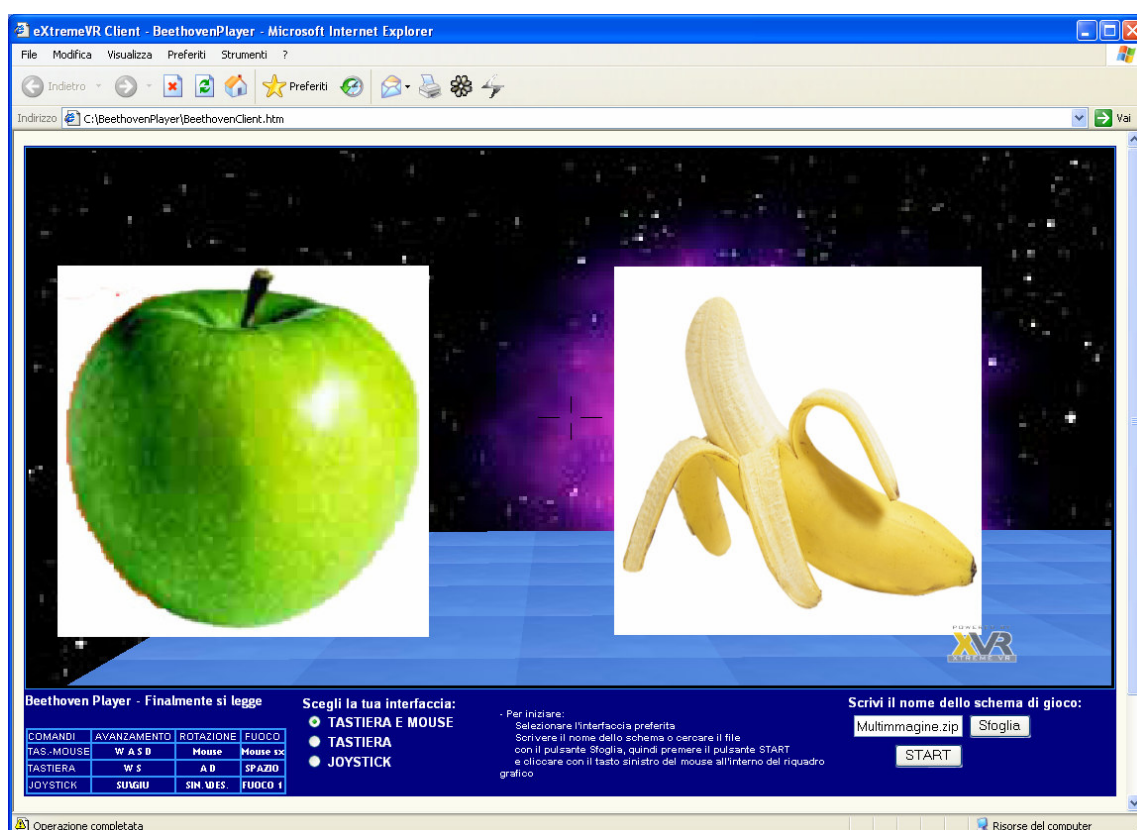
La funzione OnTimer è invocata ogni 50 millisecondi per controllare gli attivatori, ovvero le reazioni dell'ambiente all'azione del giocatore. La prima cosa da fare è memorizzare in una variabile la parola riconosciuta dal componente di riconoscimento vocale e trasmessa a mezzo VBscript. Se un attivatore prevedesse come condizione il riconoscimento di una parola, avrebbe dunque l'input già pronto all'uso. Tutte le condizioni degli attivatori vengono controllate e, in caso di verifica di una condizione, le azioni corrispondenti vengono subito eseguite. In particolare, le azioni di riproduzione suoni mettono in coda i loro suoni, che vengono mixati e riprodotti al termine del controllo di tutti gli attivatori.

Nessuno dei due cicli ha termine, perché l'applicazione non si chiude da sola. È infatti l'utente che termina l'applicazione chiudendo la finestra del browser che la ospita, ricaricando la pagina o destinando la finestra alla lettura di un altro URL.

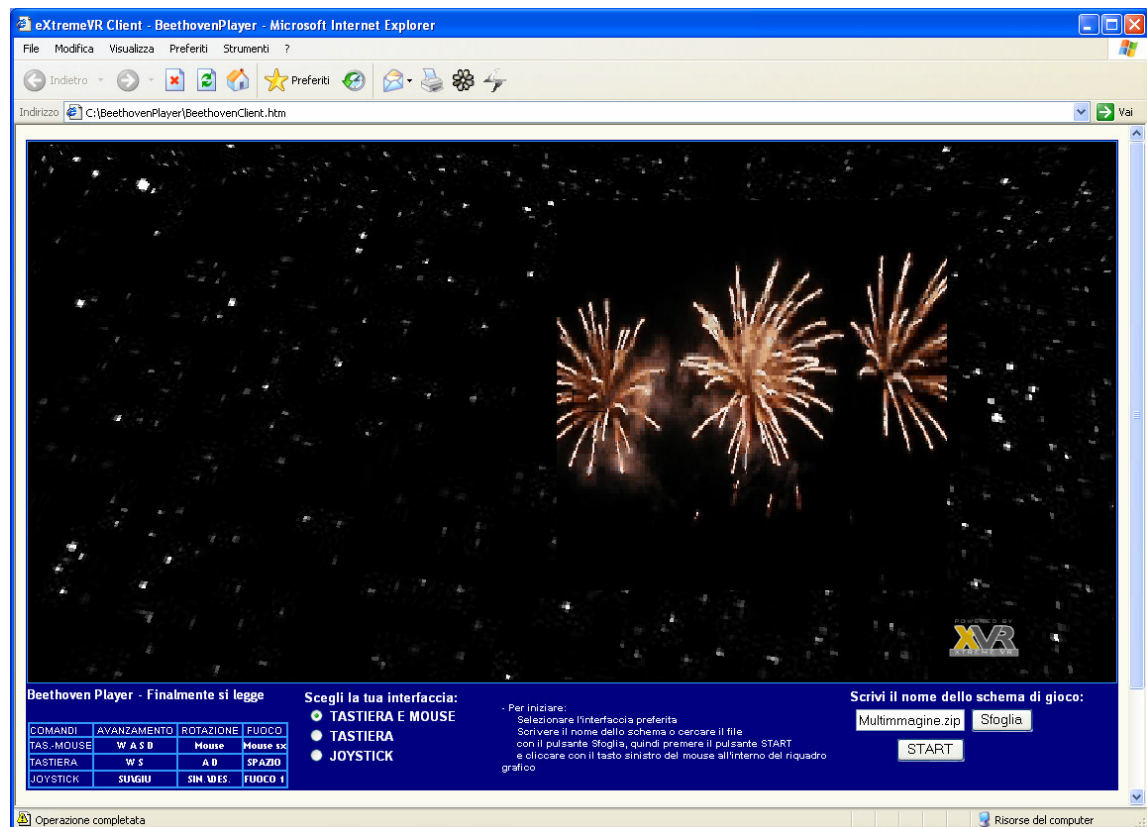
Capitolo 6: Schemi di esempio

Al pacchetto software sono acclusi alcuni schemi di esempio, con lo scopo di fungere da modelli per quattro diversi tipi di esercizio.

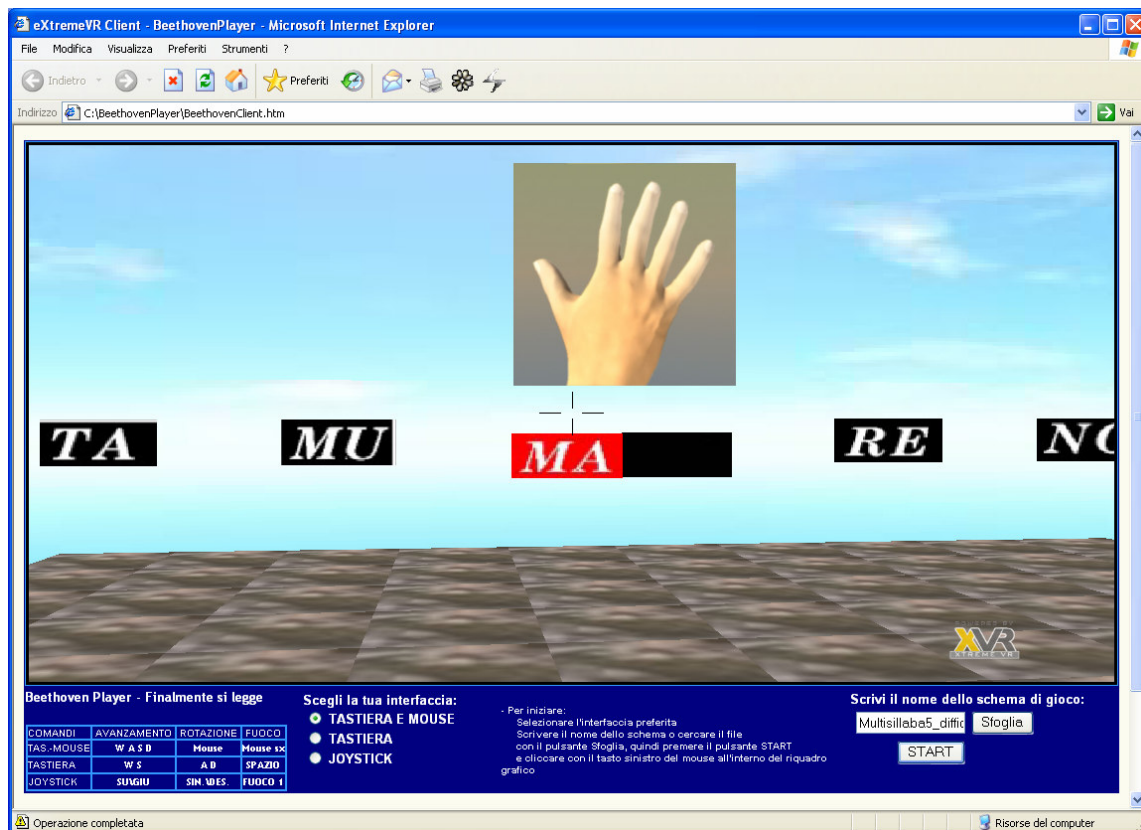
Multimmagine (sintesi sillabica): In questo esercizio, il bambino deve scegliere fra quattro figure (una mela, un nodo, una mano e una banana) quella che la voce fuori campo (dopo una breve introduzione) denomina in modo sillabato, alla velocità di circa una sillaba ogni due secondi. Ad esempio, la prima parola pronunciata è ME_LA. Per scegliere la figura il bambino deve mettere il mirino della telecamera sull'immagine e premere il tasto di fuoco. Alla pressione del tasto, se la figura è giusta, la voce pronuncia un'altra parola:



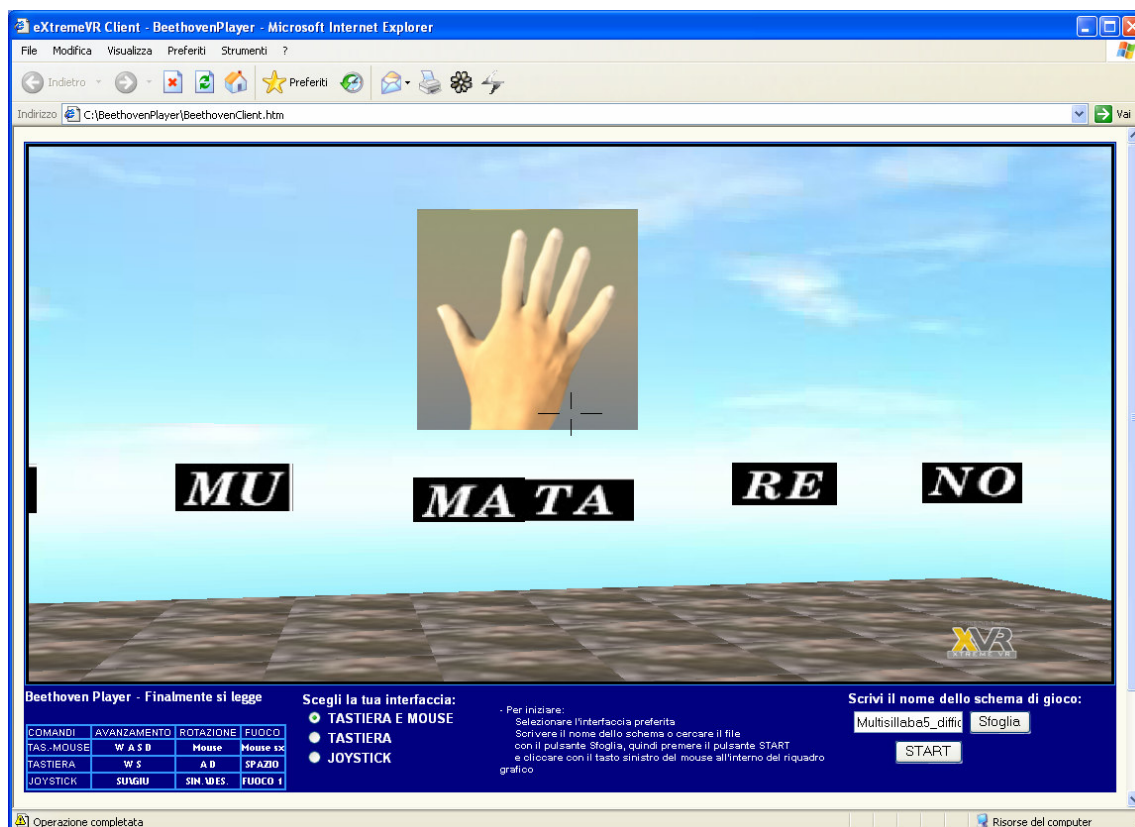
Quando il bambino ha indovinato tutte le associazioni parola-immagine, si ripete l'esercizio, ma stavolta la voce fuori campo (dopo aver annunciato questa novità al giocatore) pronuncia solo la prima sillaba dei nomi delle figure. Una volta indovinate tutte le sillabe i fuochi artificiali esplodono nel cielo stellato.



Multisillaba (completamento di parola): In questo esercizio, il giocatore si trova davanti all'immagine della mano già vista nell'esercizio Multimmagine, il cui nome è scritto sotto; il problema è che al nome manca una sillaba (al posto della sillaba c'è un riquadro vuoto)! La voce del narratore stimola il bambino a interagire con il mondo dicendo: «Questa è una MANO. Qui c'è scritto MA (la casella con MA si illumina di rosso). Che cosa manca per scrivere MA_NO (le due caselle si illuminano, prima una e poi l'altra, a tempo con la pronuncia di MA_NO)?». A destra e a sinistra della mano ci sono delle sillabe fra cui scegliere, mirando alla sillaba e premendo il tasto di fuoco.

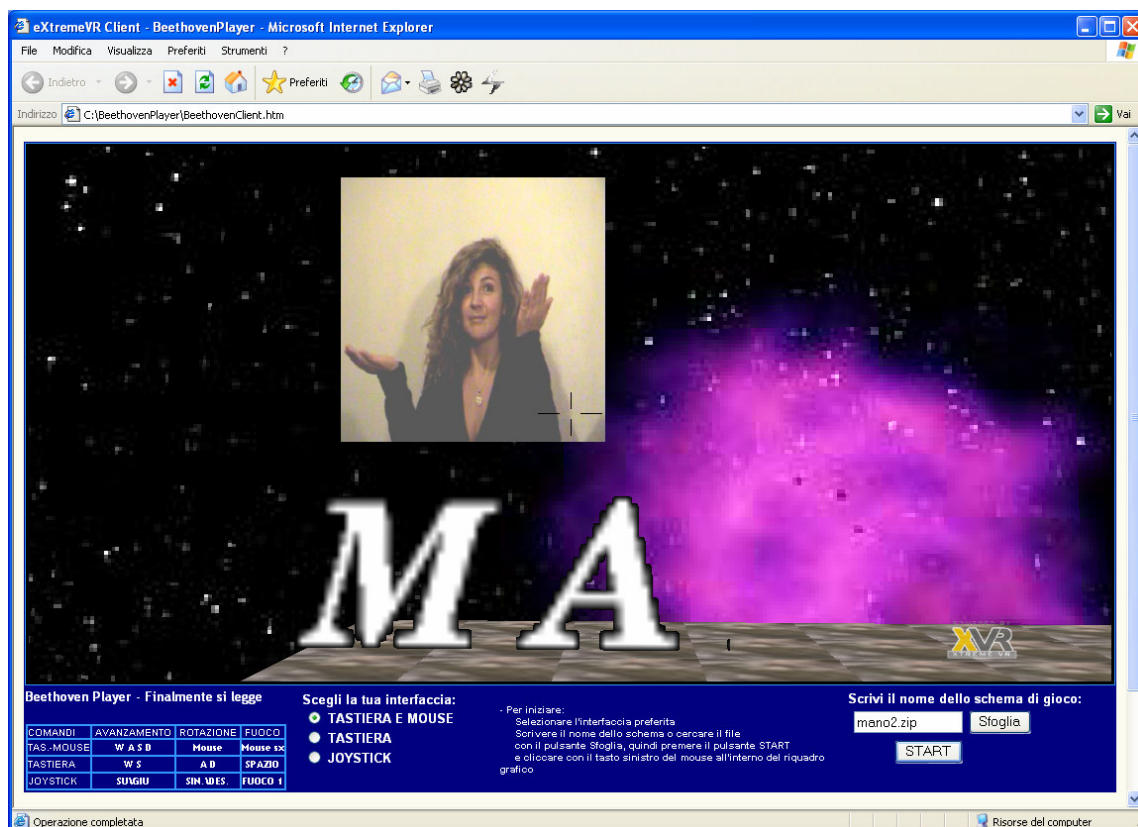


Quando il bambino sceglie una sillaba, questa compare a fianco di MA, per formare la parola. Il narratore legge la parola che si è appena formata e, se è sbagliata (per esempio MA_TA) rimuove l'ultima sillaba e dice al bambino di riprovare, se è giusta si congratula con lui per la brillante risposta!

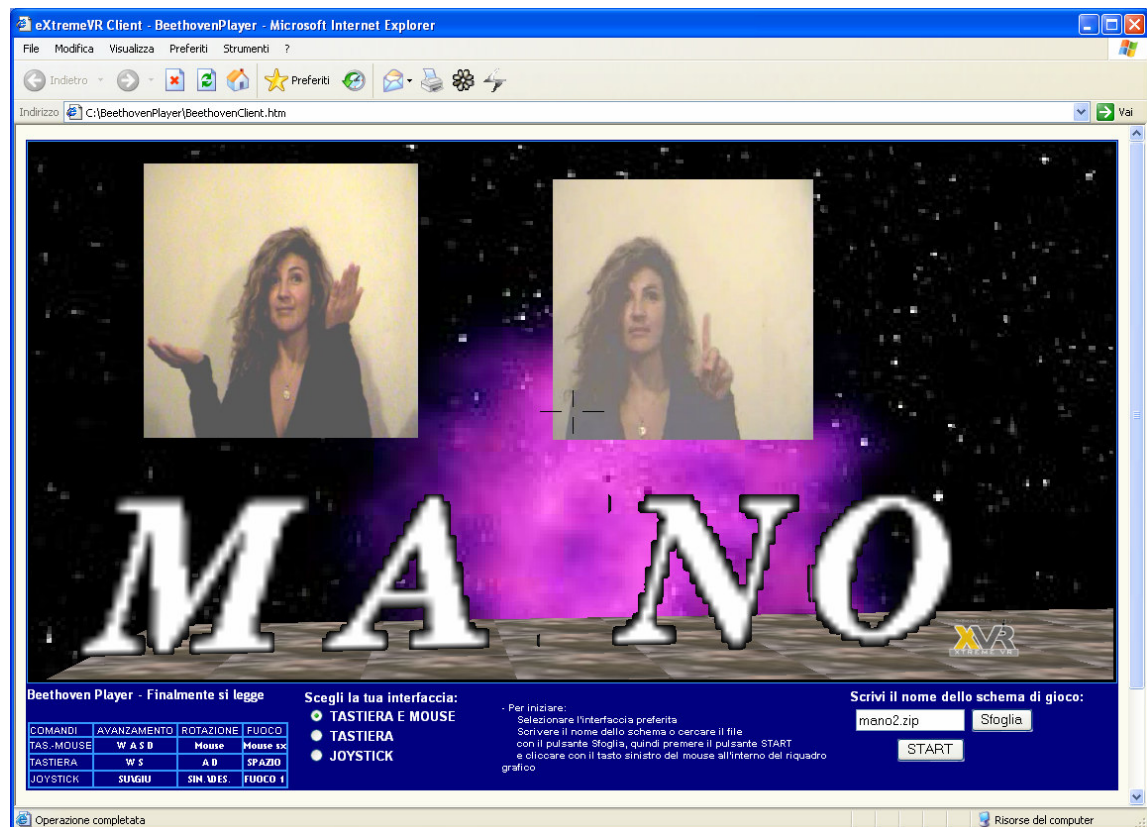


Questo esercizio esiste in due versioni, facile e difficile. Nella versione facile le sillabe disponibili sono due, in quella difficile sono quattro.

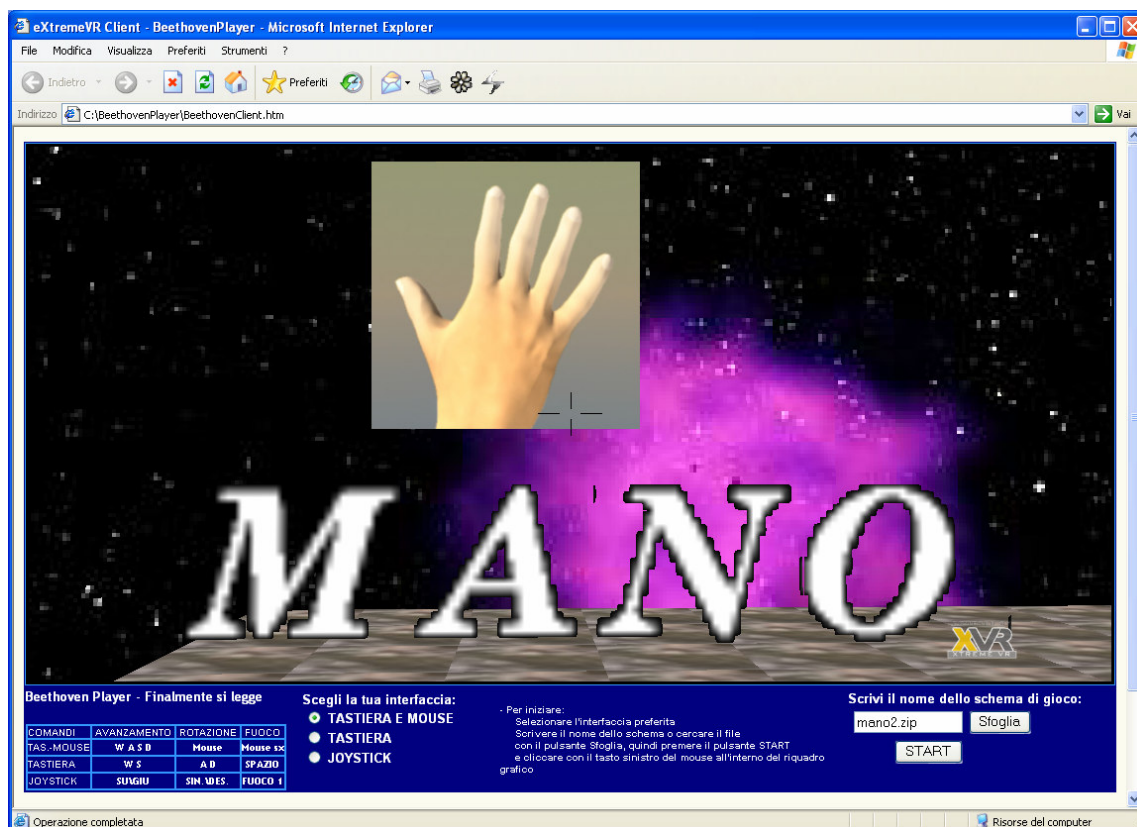
Mano (ricombinazione sillabica): In questo esercizio si parte fornendo al bambino uno stimolo: compare la sillaba MA in forma di didascalia dell'immagine di una ragazza che fa il gesto di alzare le spalle e si sente la voce della ragazza che, piena di dubbi, esclama "MA!"; in questo modo l'associazione suono/parola/immagine è ancora più intuitiva. Sillaba e immagine scompaiono dopo tre secondi e ricompaiono un secondo dopo, ripetendo lo stimolo. Poi scompaiono e ricompaiono ancora. Dopo il terzo stimolo, quando la voce della ragazza si è già spenta, il narratore (una voce maschile) chiede al bambino di leggere la sillaba.



A questo punto tocca all'insegnante fare da giudice: quando sente il bambino leggere correttamente "MA" può far proseguire l'esercizio premendo il tasto P. La parola/immagine scompare e comincia un secondo stimolo, anch'esso ripetuto tre volte: un breve filmato in cui la ragazza fa di no muovendo l'indice e dice "NO" (con la didascalia NO). Alla fine il narratore chiede al bambino anche questa sillaba e ancora una volta tocca all'insegnante dare la sua conferma della buona pronuncia con il tasto P, al che compaiono contemporaneamente le sillabe/immagine "MA" e "NO", senza suono, leggermente distanziate.

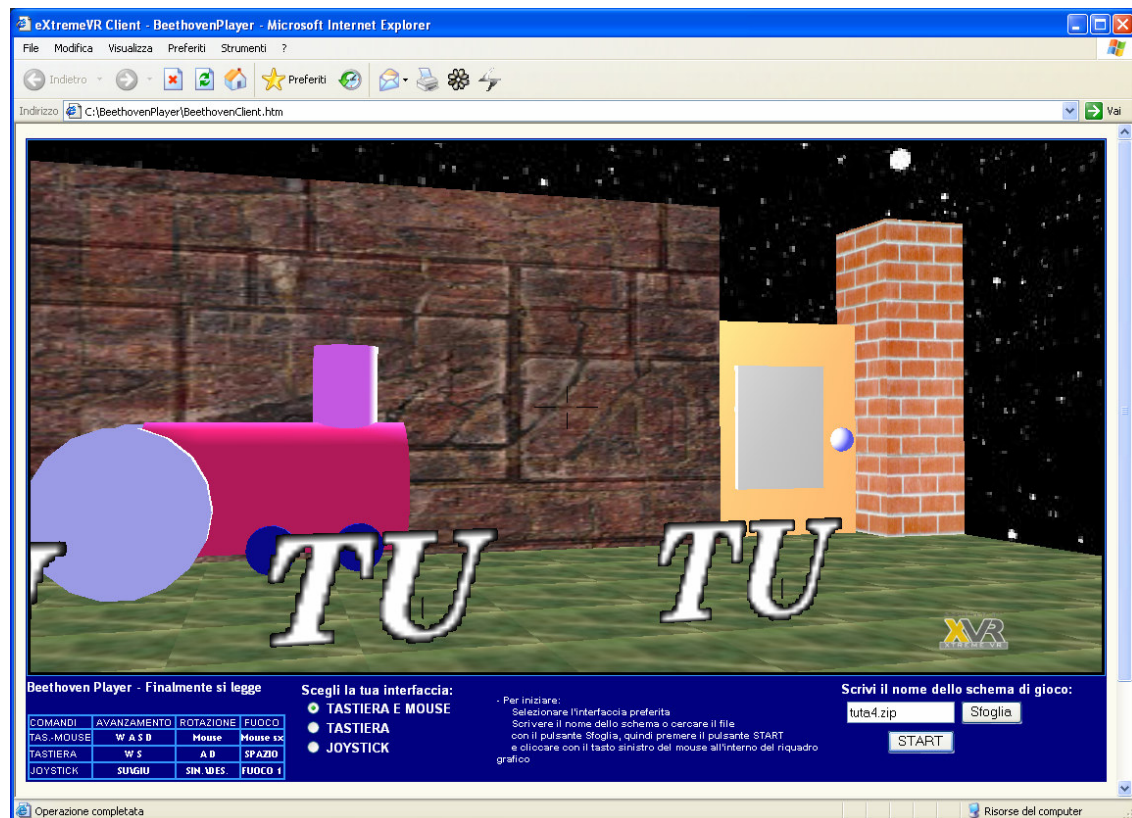


Il narratore chiede al bambino di mettere insieme le due sillabe. Questo è il momento clou dell'esercizio: il bambino deve pronunciare i due suoni uno dopo l'altro, producendo una nuova parola, "MANO"! Sentendo questa nuova parola (è sempre l'insegnante che fa proseguire il gioco premendo P) le due sillabe/immagine scompaiono e compare l'immagine della mano con la didascalia MANO e le congratulazioni del narratore.

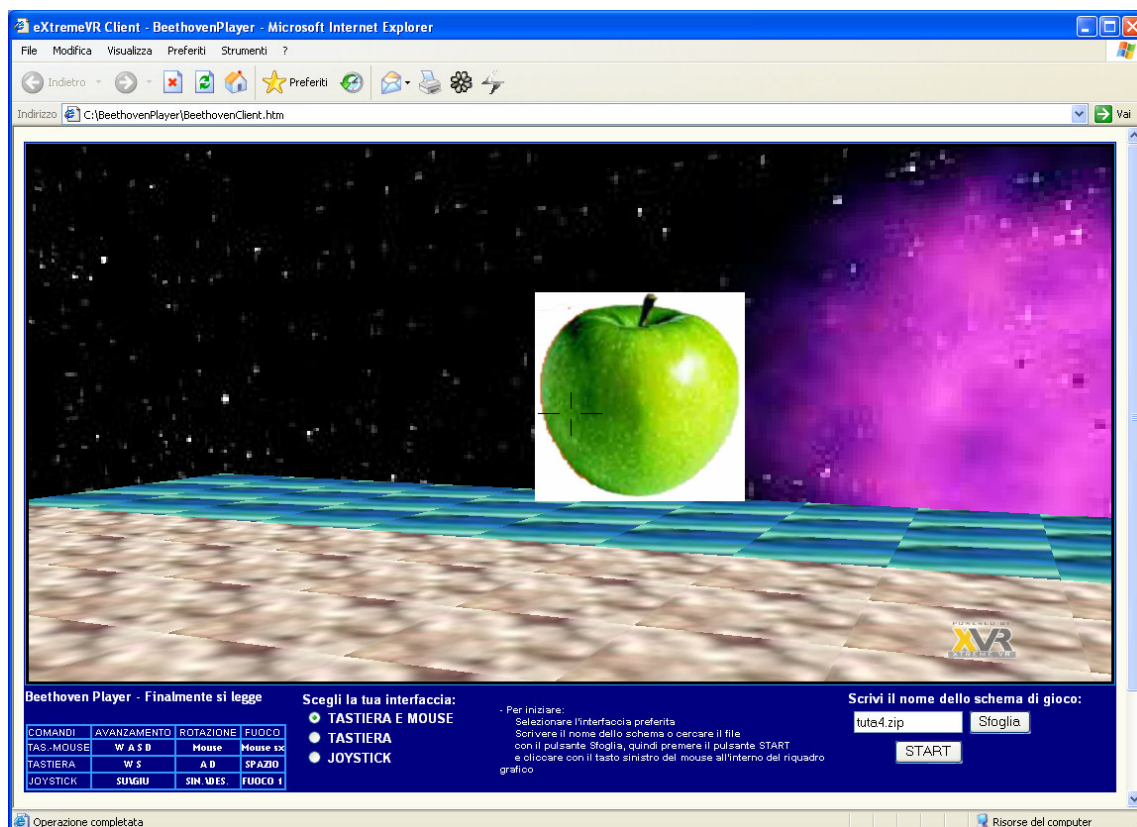


Dopo qualche attimo i fuochi d'artificio nel cielo festeggiano il bravo lettore e gli "occhi" del protagonista si alzano verso il cielo per godersi lo spettacolo.

Tuta (ricombinazione sillabica + individuazione prima sillaba): Questo schema è diverso dagli altri, nel fatto che è un accenno alla possibilità di costruire un percorso in cui si devono usare abilità differenziate, in modo più simile ad un videogioco che ad un gioco didattico. Si parte con un esercizio di ricombinazione sillabica, più difficile di quello della mano: c'è il modello tridimensionale di un treno e le sillabe TU TU TU scritte sotto, senza che alcun aiuto sonoro sia fornito al giocatore, eccetto la voce guida del narratore che gli chiede di leggere la sillaba. Il bambino dovrebbe già essersi reso conto che stavolta il mondo di gioco non può essere liberamente esplorato, dato che un muro di pietra e una colonna di mattoni gli impediscono di uscire dalla "stanzetta" in cui si trova. L'unica via d'uscita sembra essere rappresentata da una porta, ma per ora non c'è modo di aprirla.



L'insegnante e il narratore sono comunque lì ad aspettare di sentirlo leggere, perciò si va avanti con l'esercizio. Al TU del bambino (confermato dall'insegnante con il tasto 1) il treno e le sue sillabe scompaiono e compare l'immagine di un pugno che sbatte sul tavolo, con le sillabe TA TA TA. Ancora una volta il bambino legge TA (confermato con il tasto 2) e si trova davanti le sillabe TU e TA e il narratore che gli chiede di metterle insieme. Quando il bambino legge TUTA (tasto 3 per l'insegnante) compare l'immagine di una tuta e la didascalia TUTA, con annesse congratulazioni dell'insegnante e del narratore, che apre la porta e invita il bambino ad attraversarla. Al di là c'è una spiaggia con l'acqua, ma lo aspetta anche un altro esercizio! Il narratore chiede al bambino di pronunciare la prima sillaba del nome delle immagini che compariranno.



Ad ogni immagine (le immagini sono “banana”, “nodo”, “mela” e “mano”) il computer attende che il bambino pronunci una sillaba prima di mostrare l’immagine successiva (l’insegnante conferma le quattro sillabe con i tasti 4, 5, 6, 7). Infine il narratore di congratula con il giocatore, che per aver superato un esercizio così difficile ha meritato almeno un gelato (il cui acquisto è delegato all’insegnante).

Fare altri schemi simili è spaventosamente facile: ad esempio, per sostituire le immagini che raffigurano le sillabe nello schema di ricombinazione sillabica è sufficiente aprire l’editor e cambiare le definizioni degli oggetti con il gestore degli oggetti. Tradotto in clic del mouse, si tratta di cliccare sul pulsante “modifica” e cercare tramite la finestra di dialogo Sfoglia (identica a quella di Windows) l’immagine che si vuole usare, salvare e giocare. Anche cambiare i suoni è semplicissimo: anche qui, basta aprire il gestore degli attivatori, cliccare sul collegamento HTML che corrisponde al nome del file WAV da sostituire e cercare quello di rimpiazzo attraverso la solita finestra Sfoglia. Per aggiungere i fuochi d’artificio alla fine dello schema a due stanze (ricombinazione + individuazione prima sillaba) basta definire un oggetto-filmato con il gestore di oggetti e aggiungere un attivatore che lo faccia comparire alla pressione di un tasto con il gestore di attivatori. Questo trasforma in pochi minuti il terapeuta da un mero utilizzatore di

esercizi in formato software sviluppati da altri a un insieme di progettista di esercizi, implementatore di schemi, tester e utilizzatore finale. La trasformazione in implementatore è la più importante: esaminando gli schemi con i miei relatori, mi capitava spesso di sentire commenti del tipo: «lo schema va bene, ma la sillaba NO dovrebbe essere un po' più piccola» (o un po' più a destra, o dovrebbe comparire un po' più tardi). Con l'editor, il terapeuta non deve interpellare uno sviluppatore per cambiare le dimensioni dei caratteri né deve rassegnarsi ad utilizzare lo schema così com'è anche se vorrebbe modificarlo, può farlo da sé e subito, con una manciata di clic del mouse!

La mia intenzione è proprio quella di stimolare i terapeuti a sperimentare l'utilizzo del software e a fare da sé gli schemi, senza dover trasmettere le loro competenze a un progettista di schemi. Per questo non ho allegato altri schemi, perché non sia possibile prendere il pacchetto software così com'è, giocare gli schemi già presenti, abbandonarlo e cercare un altro programma didattico¹⁶.

¹⁶ In realtà, c'è anche un altro motivo: l'ingombro. Gli schemi di esempio sono molto ingombranti (anche 4 Megabyte l'uno) perché ho usato suoni digitali di qualità CD, filmati Avi e texture di alta qualità. Essendo il progetto pensato per una distribuzione attraverso Internet, le dimensioni contenute sono un fattore cruciale per assicurarne la diffusione.

Capitolo 7: Risultati e conclusioni

La forte somiglianza del mondo di gioco con alcuni fra i più noti videogiochi con visuale in prima persona e l'uso di una periferica tipicamente ludica come il joystick stimola nei bambini (soprattutto nei piccoli videogiocatori) un senso di familiarità con qualcosa a loro gradito, il che, combinato con il suggestivo contorno grafico (il cielo azzurro, le stelle o una foresta di eucalipti) invoglia a giocare con il player.

Per quanto riguarda gli insegnanti, l'utilizzo dell'editor riesce agevole solo se l'operatore possiede una alfabetizzazione informatica di base e se ha dimestichezza con i comuni programmi per ufficio di Windows¹⁷; questi pertanto sono dei prerequisiti per l'utilizzo dell'editor. All'editor è comunque accluso un manuale in formato HTML (e la sua versione stampabile) che illustra in dettaglio tutte le operazioni di base per la creazione e la modifica degli schemi di gioco.

Il pacchetto si presta anche ad essere utilizzato per lavori futuri: anche se, come ho già detto, non ho volutamente incluso una serie completa di esercizi con molte figure e parole diverse, potrebbe essere utile produrre una suite di schemi o un corso completo (come quelli descritti al capitolo 2) da offrire agli utenti; particolarmente utile sarebbe una serie di "templates", come quelli già inclusi, che implementino nuovi tipi di esercizi e che siano facilmente personalizzabili, oppure un gioco più ludico che didattico, da proporre agli studenti oltre i cinque anni.

Le due applicazioni offrono anche possibilità di espansione e perfezionamento: nel capitolo 4 avevo accennato al riconoscimento vocale. Per esempio sarebbe interessante addestrare adeguatamente un componente per il riconoscimento vocale perché riconosca le voci dei bambini e inserirlo nel player, dove troverebbe un'architettura software già disposta ad accoglierlo.

Un altro modo di rendere più amichevole l'interfaccia dell'editor sarebbe quello di visualizzare l'anteprima degli oggetti nel gestore degli oggetti (che è già predisposto ad ospitare la finestra di anteprima). L'anteprima dovrebbe dunque disporre di un parser per leggere i file AAM dei modelli 3D e di un componente per riprodurre i filmati AVI (per le immagini c'è già la libreria QT che provvede al caricamento ed alla visualizzazione, e anche il font 3D degli oggetti testo è disegnato su un file grafico PNG).

¹⁷ Vale anche per i programmi per ufficio di Linux o MacOS, che hanno le stesse metafore operative (Apri file, Salva file, Sfoglia per cartelle eccetera)

Bibliografia

- [**Bilac, Baldwin, Tanaka**] Slaven Bilac, Timothy Baldwin, Hozumi Tanaka, *Improving Dictionary Accessibility by Maximizing Use of Available Knowledge*, 2003
- [**Bishop&Clarkson**] Dorothy Bishop, Barbara Clarkson, *Written language as a window into residual language deficits: a study of children with persistent and residual speech and language impairments*, 2003
- [**Brizzolara et al.**] Daniela Brizzolara, Claudia Casalini, Barbara Sbrana, Anna Maria Chilosi, Paola Cipriani, *Memoria di lavoro fonologica e difficoltà di apprendimento della lingua scritta nei bambini con disturbo specifico del linguaggio*, Psicologia clinica dello sviluppo, Anno III numero 3, Dicembre 1999
- [**Casalini et al.**] C. Casalini, D. Brizzolara, A. Chilosi, P. Cipriani, S. Marcolini, C. Pecini, S. Roncoli, C. Burani, *Non-word repetition in children with Specific Language Impairment: a deficit in phonological working memory or in long-term verbal knowledge?*, Cortex (in stampa)
- [**Caselli&Capirci**] Maria Cristina Caselli, Olga Capirci, *Identificazione precoce di bambini con ritardo di linguaggio: metodi e strumenti di osservazione*, Atti del I convegno nazionale sull'epidemiologia dei disturbi specifici di linguaggio in età evolutiva, 2002
- [**Caselli et al.**] M. Cristina Caselli, Enrica Mariani, Manuela Pieretti, *Logopedia in età evolutiva*, Edizioni del Cerro 2003
- [**Celi, Alberti, Laganà**] Fabio Celi, Claudia Alberti, Maria Rita Laganà, *Avviamento alla lettura*, Erickson 1995
- [**Celi&Romani**] Fabio Celi, Francesco Romani, *Macchine per imparare*, Erickson 1997
- [**Chilosi et al. 1998**] Anna Maria Chilosi, Paola Cipriani, Antonella Giorgi, Lucia Pfanner, Barbara Sbrana, Piero Bottari, *Follow-up di bambini affetti da Disturbo Specifico del Linguaggio*, Psicologia clinica dello sviluppo, Anno II numero 3, Dicembre 1998
- [**Chilosi et al. 2003**] Anna Maria Chilosi, Laura Lami, Claudia Pizzoli, Barbara Pignatti, Daniela D'Alessandro, Barbara Gruppioni, Paola Cipriani, Daniela Brizzolara, *Profili neuropsicologici nella dislessia evolutiva*, Psicologia clinica dello sviluppo, Anno VII numero 2, Agosto 2003
- [**Cipriani et al.**] P. Cipriani, A. M. Chilosi, T. Fapore, L. Pfanner, C. Rasi, *Studio clinico epidemiologico dei disturbi specifici di linguaggio in età prescolare e scolare*,

Atti del I convegno nazionale sull'epidemiologia dei disturbi specifici di linguaggio in età evolutiva, 2002

[Emiliani, Furia, Superchi] M. Emiliani, G. Furia, F. Superchi, *Ipotesi per l'utilizzo di un metodo sillabico di alfabetizzazione con bambini a rischio di disturbi di apprendimento del linguaggio scritto*

[Fabbro] Franco Fabbro, *Il cervello bilingue*, 1996

[Judica, Cucciaioni, Verni] Anna Judica, Cecilia Cucciaioni, Francesca Verni, «*Ma io non so leggere*». *Procedure e materiali per l'intervento riabilitativo di bambini di scuola elementare con disturbo specifico della lettura*, pubblicato all'interno di [Caselli et al.]

[Leonard] Laurence B. Leonard, *Verso una spiegazione dei Disturbi di Linguaggio nei bambini: il valore dello studio crosslinguistico*, Psicologia clinica dello sviluppo, Anno II numero 3, Dicembre 1998

[Lurija] Aleksandr R. Lurija, *La comunicazione verbale*, 1978

[Mazzacurati&Rinaldi] M. S. Mazzacurati Bonsembiante, M. V. Rinaldi, *Per una teoria della riabilitazione logopedica*, 1998

[Rapin&Allen] I. Rapin, D. A. Allen, *Developmental dysphasia and autism in pre-school children, characteristics and subtypes*, Proceedings of International Symposium: Specifics Speech and Language Disorders in Children, University of Reading, 1987

[Sabbadini&Caselli] Letizia Sabbadini, Maria Cristina Caselli, *Indicazioni per la diagnosi, la valutazione e l'intervento riabilitativo dei bambini con Disturbo Specifico del Linguaggio*, Psicologia clinica dello sviluppo, Anno II numero 3, Dicembre 1998

[Stackhouse] Joy Stackhouse, *Barriers to literacy development in children with speech and language difficulties*, 2000

[Stella&Pizzoli] Giacomo Stella, Claudia Pizzoli, *Profili di Disturbo Specifico di Linguaggio e possibilità di prevedere l'evoluzione delle abilità linguistiche nel corso dello sviluppo*, Psicologia clinica dello sviluppo, Anno II numero 3, Dicembre 1998

[Tressoldi] P. E. Tressoldi, *Prevenzione e trattamento precoce delle difficoltà strumentali di lettura e scrittura*, Procedure e metodi di trattamento nei disordini della comunicazione, Atti del III corso di aggiornamento ULI, 1998

[Volterra] Virginia Volterra, *I bambini con Disturbo Specifico del Linguaggio. Presentazione*, Psicologia clinica dello sviluppo, Anno II numero 3, Dicembre 1998

[Wikipedia] *Proto-Canaanite Alphabet*, Wikipedia <http://www.wikipedia.org/>